

清華大學
Tsinghua University

USENIX ATC'25

JENGA: Enhancing LLM Long-Context Fine-Tuning with Contextual Token Sparsity

Tuowei Wang, Xingyu Chen, Kun Li, Ting Cao, Ju Ren*, Yaoxue Zhang

Tsinghua University; Microsoft Research

BACKGROUND & MOTIVATION

Enhancing Data Processing with Expanding Context Windows

- Efficient data processing has long been humanity's aspiration.
- LLMs with larger context windows are becoming integral to modern applications.

Large Language Models

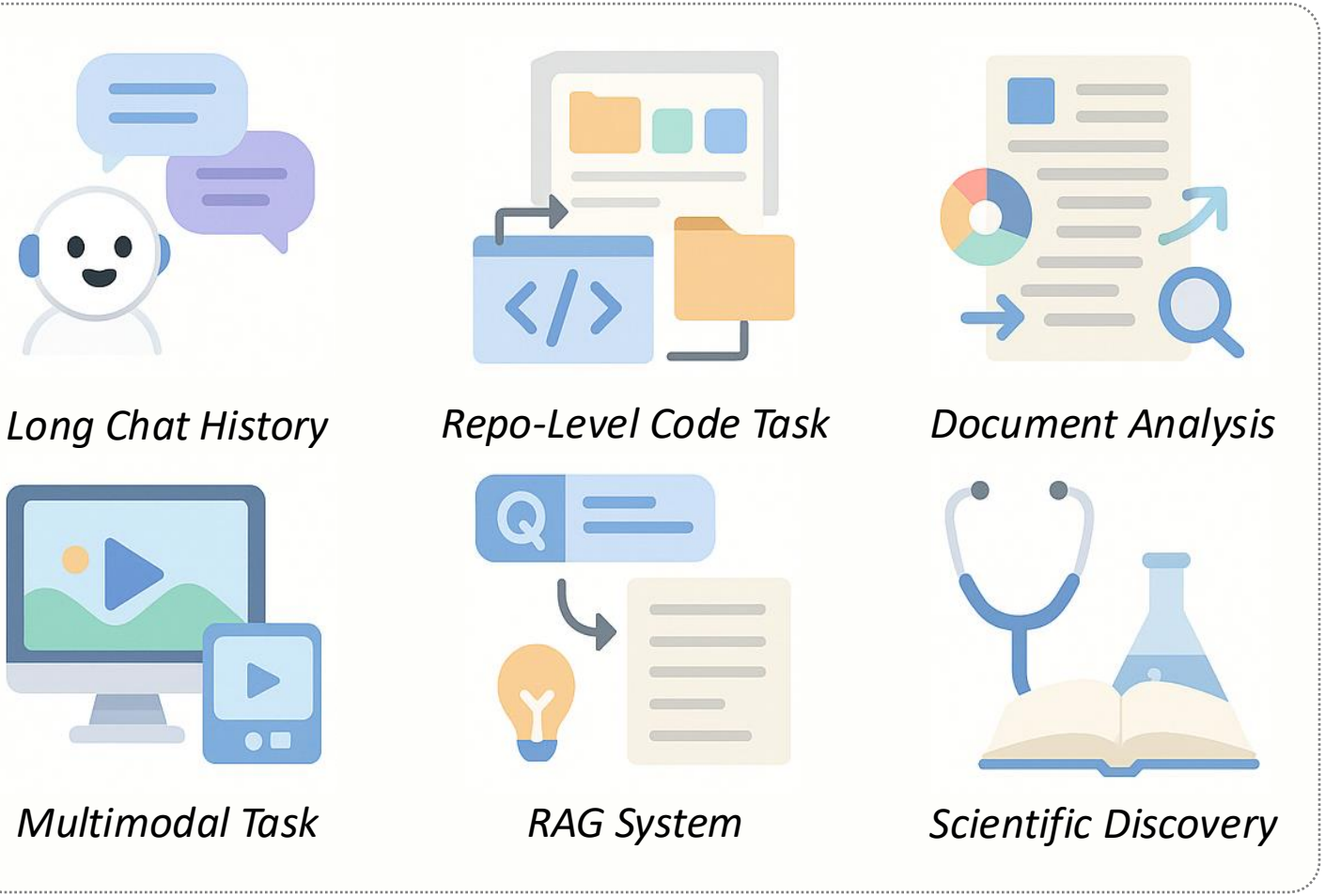
Fixed Context Window

↓

Modeling Input Sequences

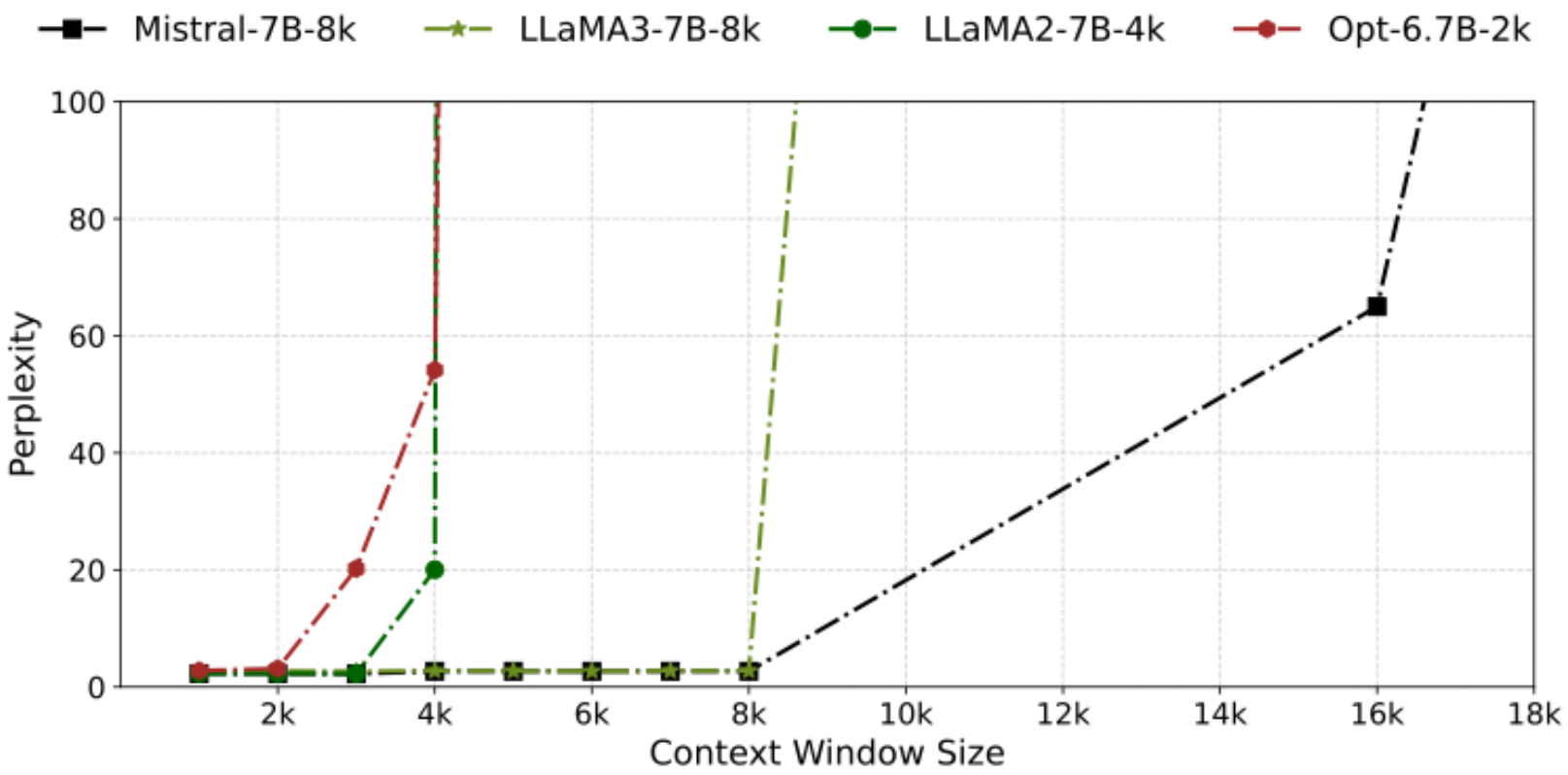
↓

Next-Token Prediction



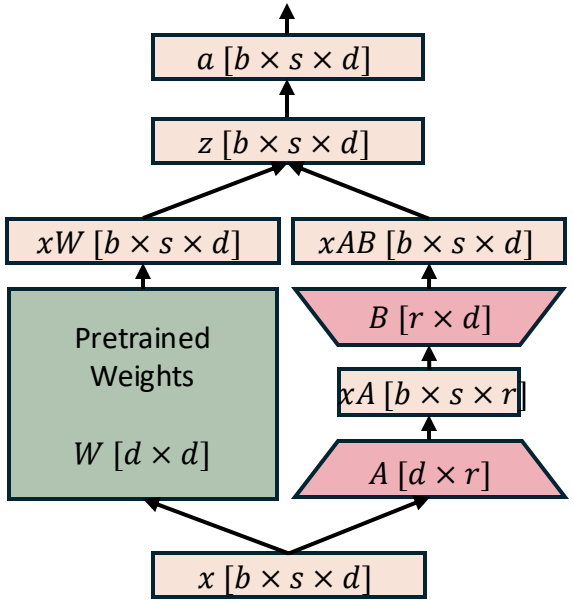
Limitations of Fixed Context Windows in LLMs

- LLMs are typically pretrained with fixed content window sizes.
  - E.g., *OPT – 2K*, *LLaMA2 – 4K*, *Mistral – 8K*, *LLaMA3 – 8K*, *DeepSeek-V3 – 128K*
- When inputs exceed these predefined limits, model performance degrades markedly.



Extending LLM Context Windows with Fine-Tuning

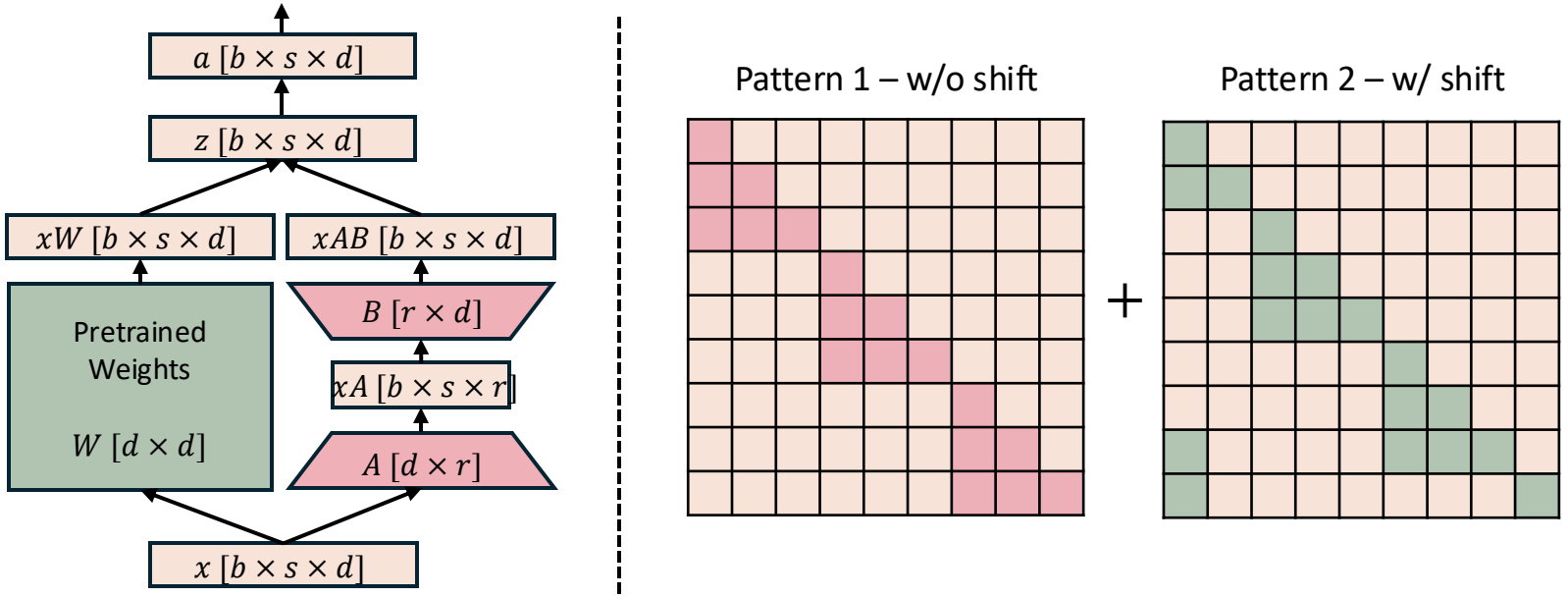
- The context window of pre-trained LLMs can be extended through fine-tuning on longer sequences.
- Two representative methods to improve the efficiency of fine-tuning:
  - LoRA: Updates only a small subset of parameters, keeping the majority of the model frozen.



LoRA

Extending LLM Context Windows with Fine-Tuning

- The context window of pre-trained LLMs can be extended through fine-tuning on longer sequences.
- Two representative methods to improve the efficiency of fine-tuning:
  - LoRA: Updates only a small subset of parameters, keeping the majority of the model frozen.
  - LongLoRA: Builds on LoRA by introducing sparsity to approximate standard dense attention.



Extending LLM Context Windows with Fine-Tuning

- The context window of pre-trained LLMs can be extended through fine-tuning on longer sequences.
- Two representative methods to improve the efficiency of fine-tuning:
  - LoRA: Updates only a small subset of parameters, keeping the majority of the model frozen.
  - LongLoRA: Builds on LoRA by introducing sparsity to approximate standard dense attention.

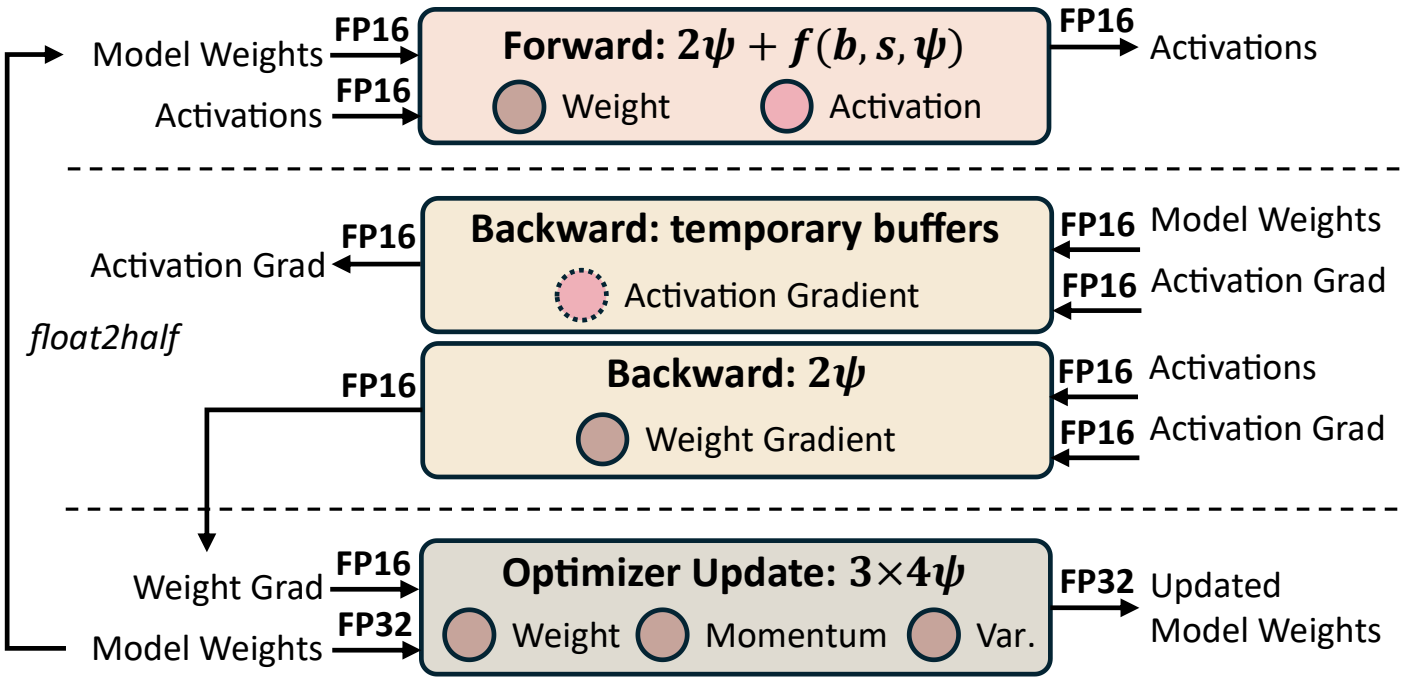
Model	Llama2-4K	Llama3-4K	Mistral-4K	OPT-4K	Llama2-8K	Llama3-8K	Mistral-8K	OPT-8K
Naïve	67.9	78.4	73.4	63.8	94.0	111.3	87.9	99.5
LoRA	39.2	43.4	39.3	36.1	73.1	79.0	60.9	65.4
LongLoRA	41.3	43.9	39.3	38.1	77.1	80.4	64.8	67.3

The Memory Bottleneck Still Remains!

INSIGHT & CHALLENGE

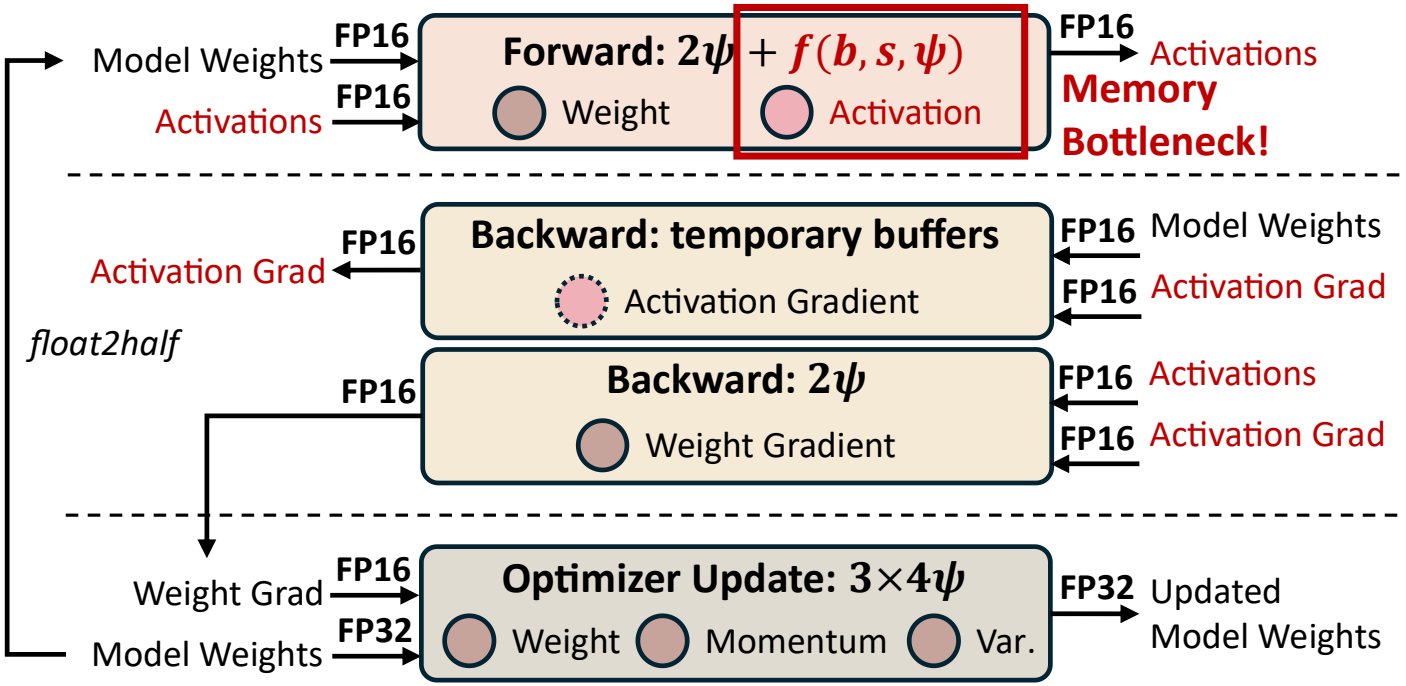
Analysis: Fine-tuning Memory Breakdown

- The memory consumption of fine-tuning consists of two parts: model states and residual states.
- Model States: Parameters, gradients, and optimizer states.
- Residual States: Activations, temporary buffers, and fragmented memory.



Analysis: Fine-tuning Memory Breakdown

- The memory consumption of fine-tuning consists of two parts: model states and residual states.
- Model States: Parameters, gradients, and optimizer states.
- Residual States: Activations, temporary buffers, and fragmented memory.

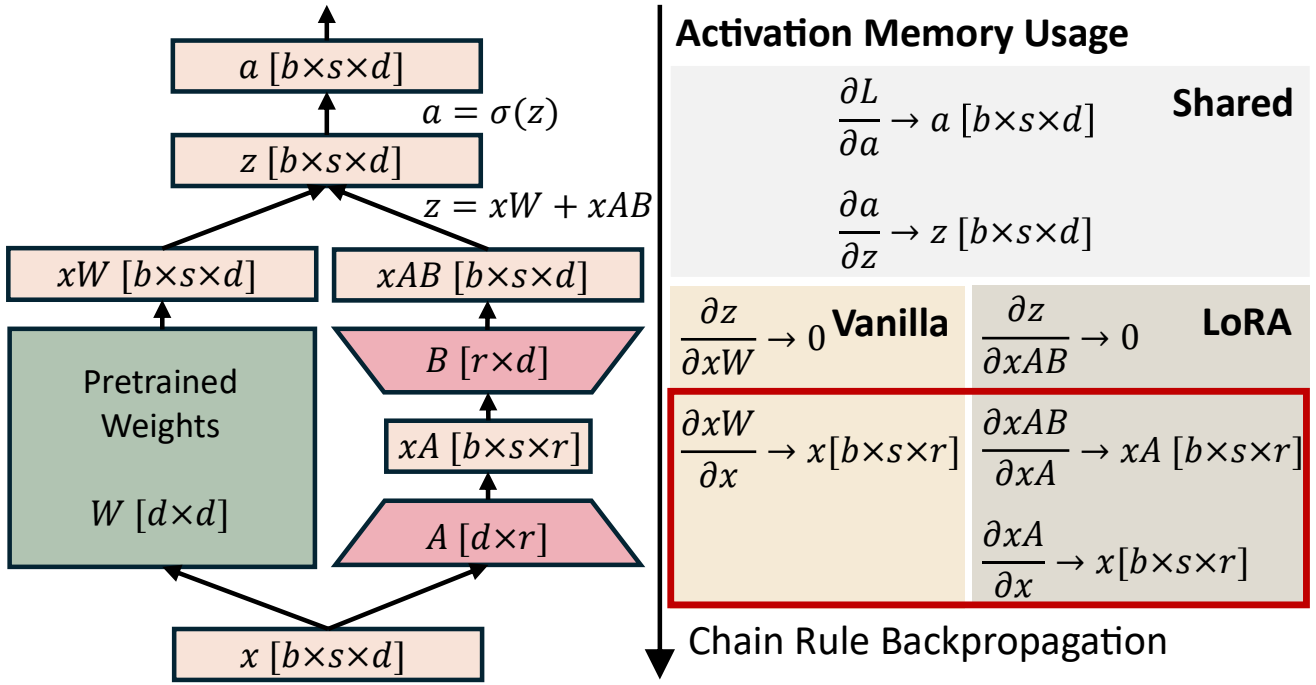


Activation memory usage (compared to model states) of GPT-3 175B.

Model States	$s = 4K$	$s = 8K$	$s = 16K$	$s = 32K$	$s = 64K$
$16 \times 175B = 2.8T$	937G	3.42T	13.0T	50.8T	201T
-	0.34×	1.22×	4.65×	18.14×	71.6×

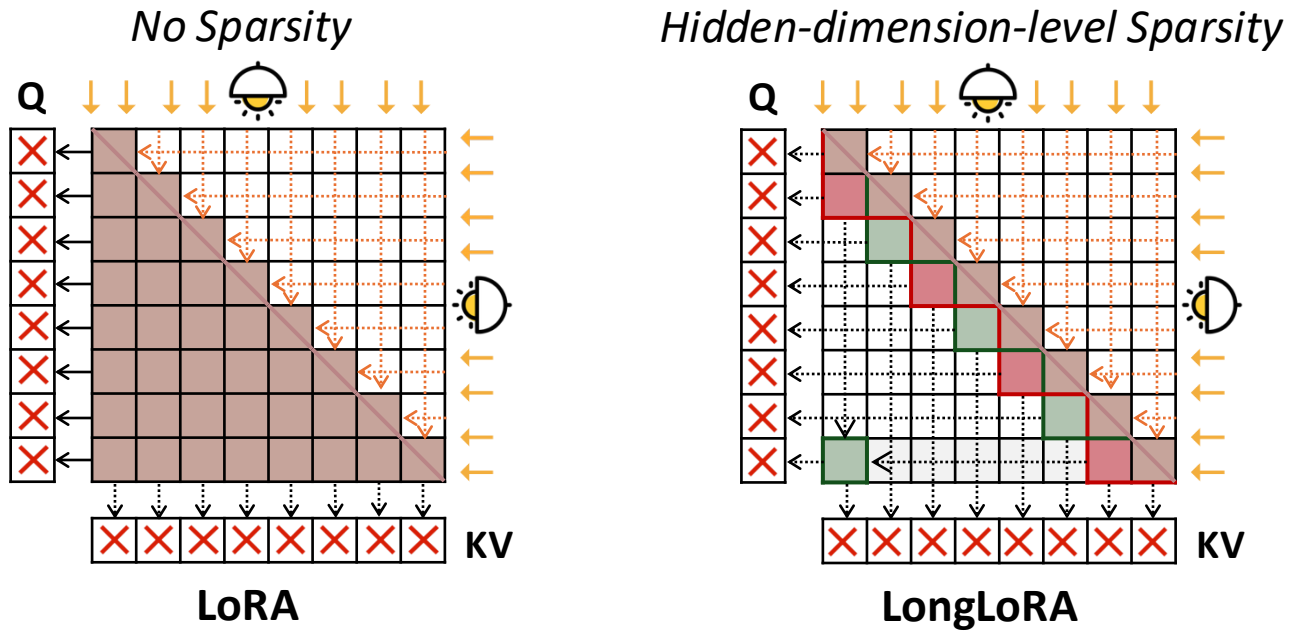
Analysis: Fine-tuning Memory Breakdown

- Representative parameter-efficient fine-tuning approach: *LoRA*.
- The memory consumption for gradients and optimizer states is reduced.
- However, LoRA not only fails to alleviate activation memory usage but increases it instead.



Analysis: Fine-tuning Memory Breakdown

- Representative sparsity-based approach: *LongLoRA*.
- Achieves computational savings compared to LoRA, but fails to provide additional memory reduction.
- Essential Reason: *Shadowy Sparsity*
  - Once a token is involved in the computation, its activations are stored regardless of its usage.



Insight: Contextual Token Sparsity

- Observation: Natural language is inherently redundant, especially in long-context scenarios.

Sparsity ratios (the proportion of values below 0.3 of the maximum) of attention scores across sequence lengths.

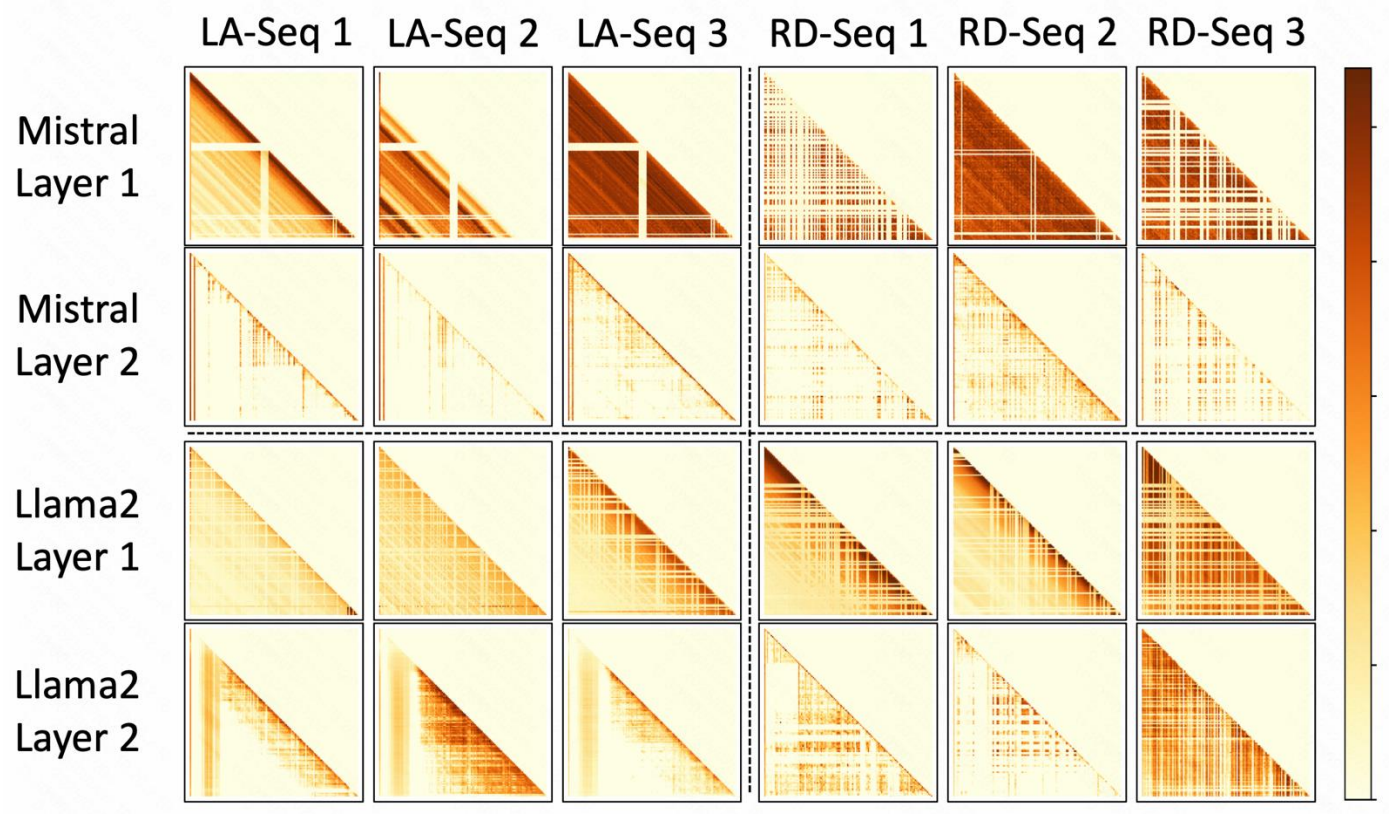
Seq len.	4K	6K	8K	10K	12K	14K	16K
Llama2	38.6%	48.8%	48.1%	65.3%	69.5%	62.2%	69.6%
Llama3	44.6%	46.0%	54.7%	52.3%	57.8%	50.2%	58.9%

Insight: Contextual Token Sparsity

- Observation: Natural language is inherently redundant, especially in long-context scenarios.
- Insight: Optimize LLM long-context fine-tuning by identifying and retaining only the most informative tokens – *Contextual Token Sparsity*.

Insight: Contextual Token Sparsity

- Observation: Natural language is inherently redundant, especially in long-context scenarios.
- Insight: Optimize LLM long-context fine-tuning by identifying and retaining only the most informative tokens – *Contextual Token Sparsity*.



Two Key Characteristics:

1. Token-wise. The token embeddings in the sequence exhibit varying levels of importance.
2. Contextual. The distribution of valuable tokens dynamically shifts based on input texts and varies across model layers.

Challenge: Identification, Detection, and Performance.

Challenge 1 – Redundant Token Identification

- Eliminating informative tokens risks degrading model performance.
- Retaining excessive tokens results in inefficient resource utilization..

Challenge: Identification, Detection, and Performance.

Challenge 1 – Redundant Token Identification

- Eliminating informative tokens risks degrading model performance.
- Retaining excessive tokens results in inefficient resource utilization.

Challenge 2 – Sparsity Pattern Detection

- The optimal sparsity patterns vary across different inputs and layers.
- The selection mechanism should incur minimal computational and memory overhead.

Challenge: Identification, Detection, and Performance.

Challenge 1 – Redundant Token Identification

- Eliminating informative tokens risks degrading model performance.
- Retaining excessive tokens results in inefficient resource utilization.

Challenge 2 – Sparsity Pattern Detection

- The optimal sparsity patterns vary across different inputs and layers.
- The selection mechanism should incur minimal computational and memory overhead.

Challenge 3 – Efficient Processing Performance

- The variability in token selection across layers incurs additional token movements.
- Naïve gradient computation of model loss amplifies activation memory usage.

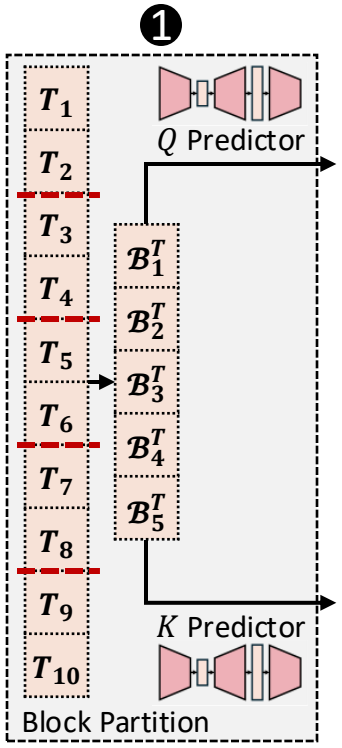
SYSTEM DESIGN

Design Overview – JENGA

- We propose JENGA, an LLM long-context fine-tuning system with contextual token sparsity.

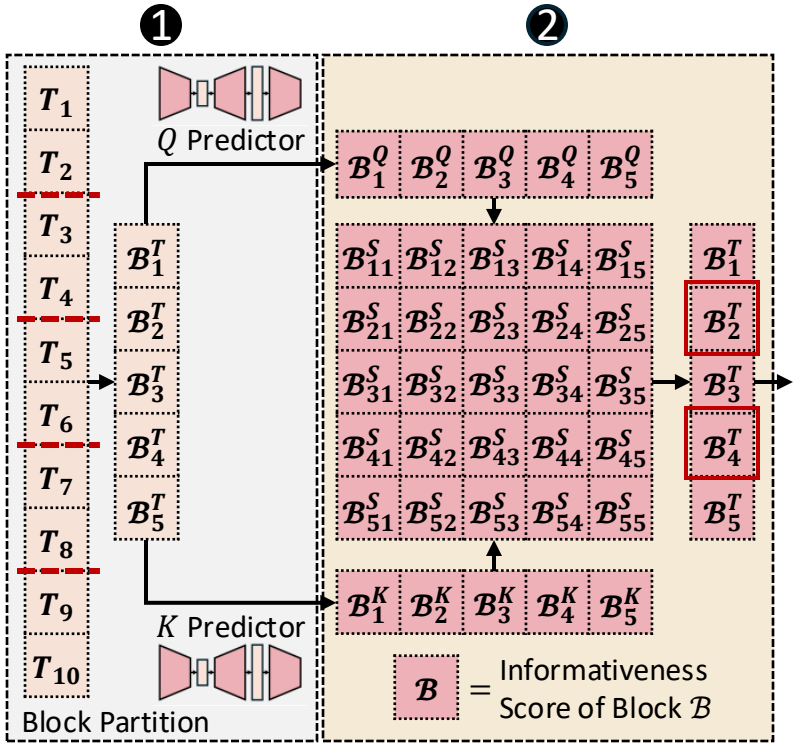
Design Overview – JENGA

- We propose JENGA, an LLM long-context fine-tuning system with contextual token sparsity.
- Step 1: Token embeddings are first partitioned into blocks and fed into the pattern predictors.



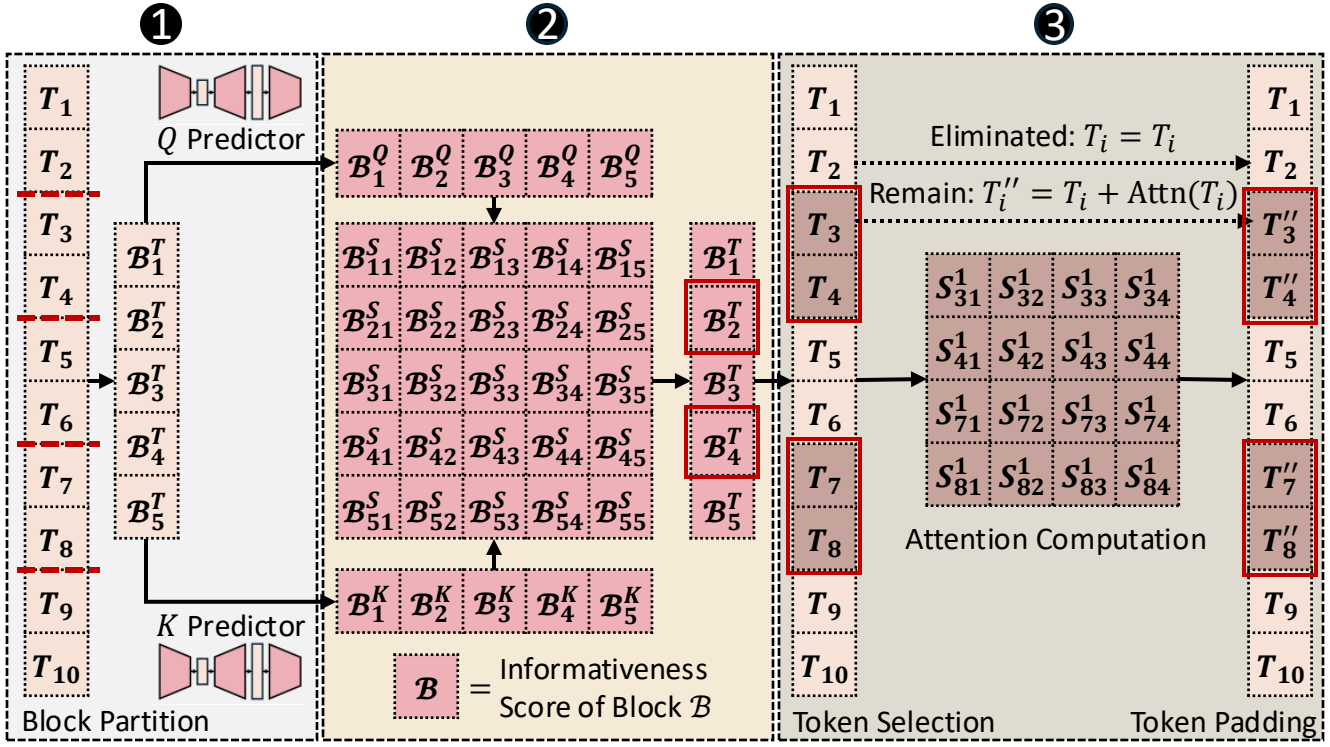
Design Overview – JENGA

- We propose JENGA, an LLM long-context fine-tuning system with contextual token sparsity.
- Step 1: Token embeddings are first partitioned into blocks and fed into the pattern predictors.
- Step 2: Token elimination algorithm effectively identifies and retains only the most informative tokens.



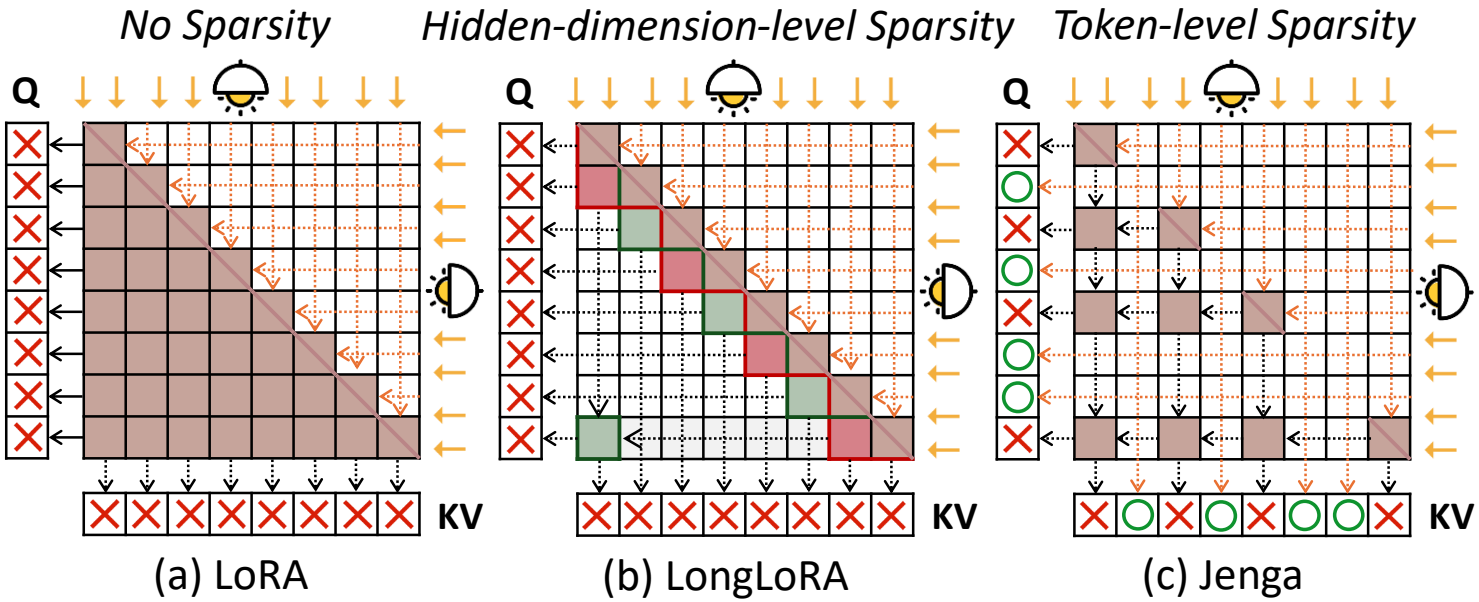
Design Overview – JENGA

- We propose JENGA, an LLM long-context fine-tuning system with contextual token sparsity.
- Step 1: Token embeddings are first partitioned into blocks and fed into the pattern predictors.
- Step 2: Token elimination algorithm effectively identifies and retains only the most informative tokens.
- Step 3: Optimized kernels efficiently perform token selection, computation, and padding.



Design Overview – JENGA

- We propose JENGA, an LLM long-context fine-tuning system with contextual token sparsity.
- Step 1: Token embeddings are first partitioned into blocks and fed into the pattern predictors.
- Step 2: Token elimination algorithm effectively identifies and retains only the most informative tokens.
- Step 3: Optimized kernels efficiently perform token selection, computation, and padding.



Design 1: Information-driven Token Elimination

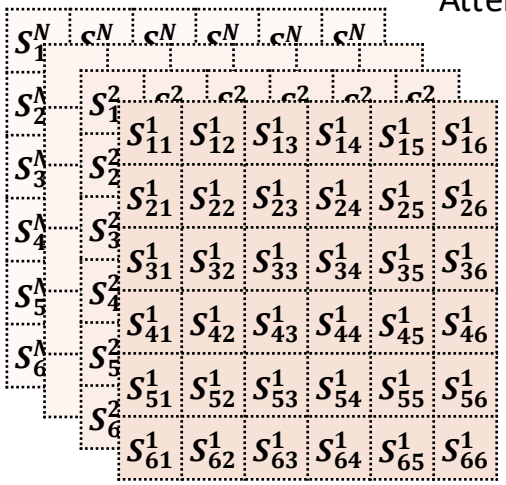
- *To fully exploit contextual token sparsity, it is essential to accurately identify the redundant tokens.*

Design 1: Information-driven Token Elimination

➢ *To fully exploit contextual token sparsity, it is essential to accurately identify the redundant tokens.*

- JENGA proposes an informative-driven algorithm that dynamically identifies and eliminates redundant tokens while preserving accuracy.

Attention Scores



Token Elimination Algorithm

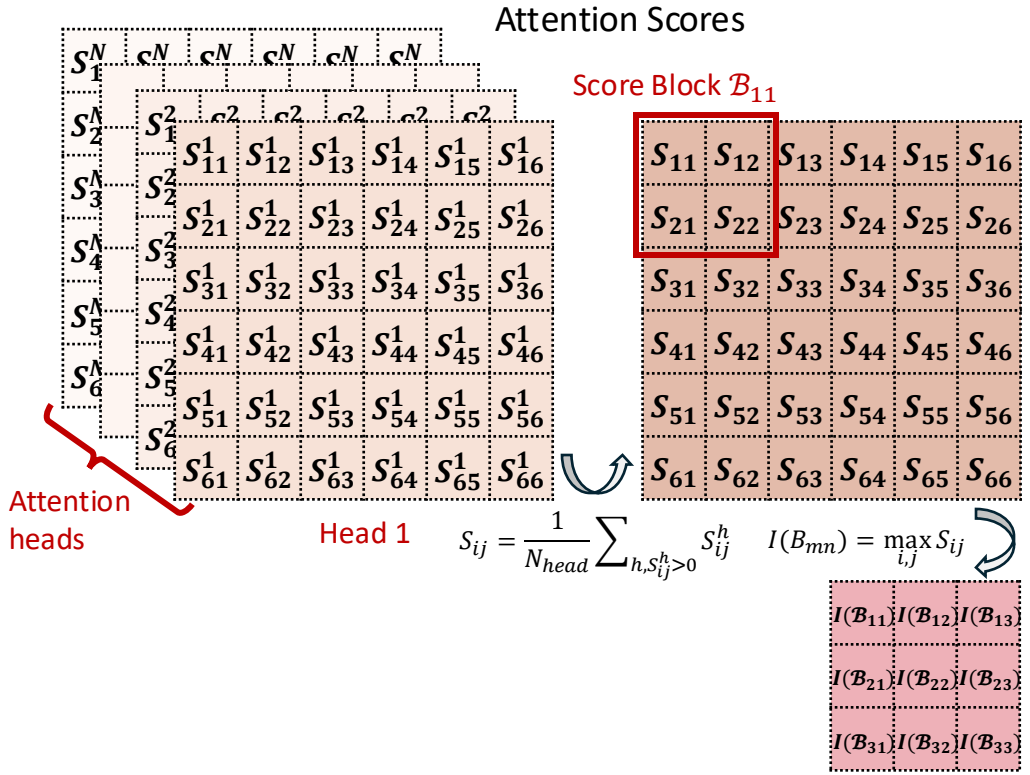
1. *Token Informativeness:*

$$I(T_j) = \sum_{i \neq j} S_{ij} = \sum_{i \neq j} Q_i K_j$$

Design 1: Information-driven Token Elimination

➢ *To fully exploit contextual token sparsity, it is essential to accurately identify the redundant tokens.*

- JENGA proposes an informative-driven algorithm that dynamically identifies and eliminates redundant tokens while preserving accuracy.



Token Elimination Algorithm

1. *Token Informativeness:*

$$I(T_j) = \sum_{i \neq j} S_{ij} = \sum_{i \neq j} Q_i K_j$$

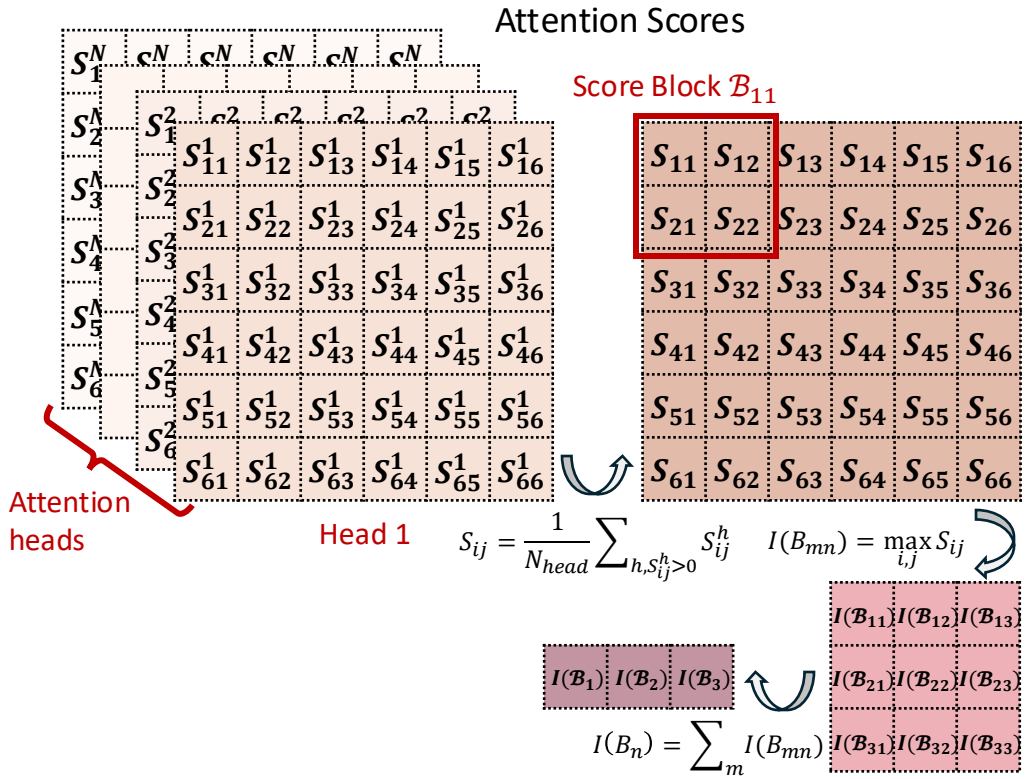
2. *Score Block Informativeness:*

$$I(\mathcal{B}_{mn}^S) = \max_{S_{ij} \in \mathcal{B}_{mn}^S} S_{ij} = \max_{S_{ij} \in \mathcal{B}_{mn}^S} \frac{\sum_{h,S_{ij}^h>0} S_{ij}^h}{N_{\text{head}}}$$

Design 1: Information-driven Token Elimination

➢ *To fully exploit contextual token sparsity, it is essential to accurately identify the redundant tokens.*

- JENGA proposes an informative-driven algorithm that dynamically identifies and eliminates redundant tokens while preserving accuracy.



Token Elimination Algorithm

1. *Token Informativeness:*

$$I(T_j) = \sum_{i \neq j} S_{ij} = \sum_{i \neq j} Q_i K_j$$

2. *Score Block Informativeness:*

$$I(\mathcal{B}_{mn}^S) = \max_{S_{ij} \in \mathcal{B}_{mn}^S} S_{ij} = \max_{S_{ij} \in \mathcal{B}_{mn}^S} \frac{\sum_{h,S_{ij}^h>0} S_{ij}^h}{N_{\text{head}}}$$

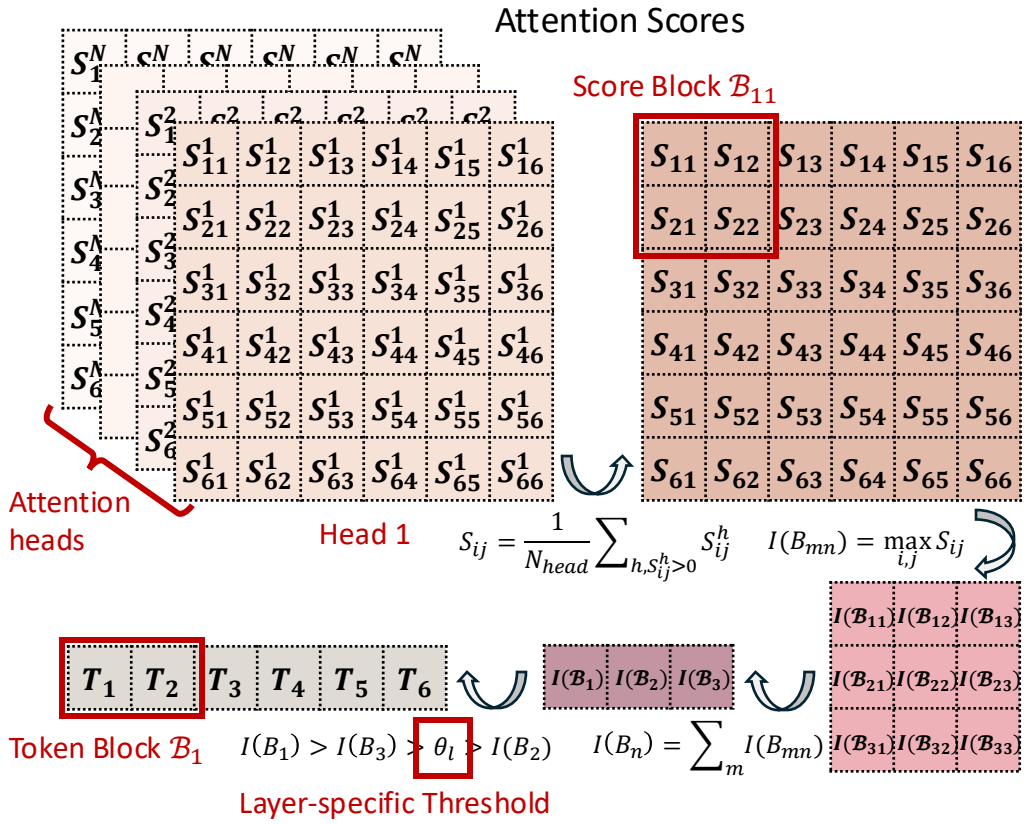
3. *Token Block Informativeness:*

$$I(B_n) = \sum_m I(B_{mn})$$

Design 1: Information-driven Token Elimination

➢ *To fully exploit contextual token sparsity, it is essential to accurately identify the redundant tokens.*

- JENGA proposes an informative-driven algorithm that dynamically identifies and eliminates redundant tokens while preserving accuracy.



Token Elimination Algorithm

1. *Token Informativeness:*

$$I(T_j) = \sum_{i \neq j} S_{ij} = \sum_{i \neq j} Q_i K_j$$

2. *Score Block Informativeness:*

$$I(\mathcal{B}_{mn}^S) = \max_{S_{ij} \in \mathcal{B}_{mn}^S} S_{ij} = \max_{S_{ij} \in \mathcal{B}_{mn}^S} \frac{\sum_{h,S_{ij}^h>0} S_{ij}^h}{N_{\text{head}}}$$

3. *Token Block Informativeness:*

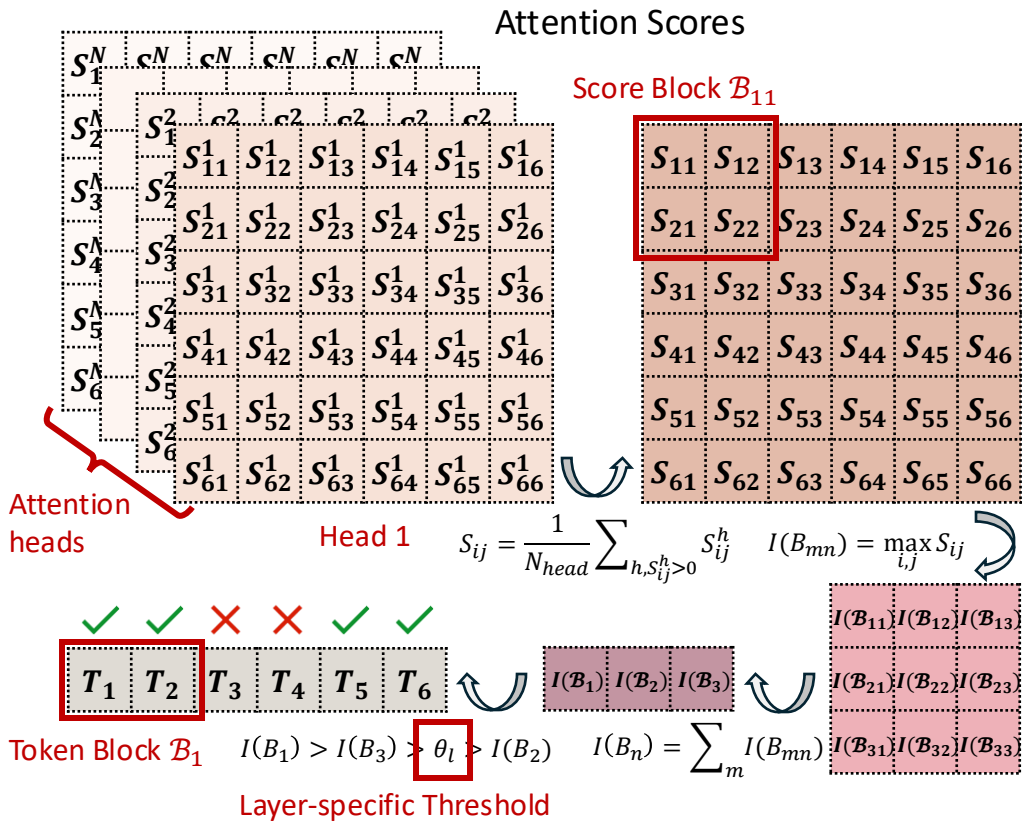
$$I(B_n) = \sum_m I(B_{mn})$$

4. *Compare with layer-specific threshold θ_l.*

Design 1: Information-driven Token Elimination

➢ *To fully exploit contextual token sparsity, it is essential to accurately identify the redundant tokens.*

- JENGA proposes an informative-driven algorithm that dynamically identifies and eliminates redundant tokens while preserving accuracy.



Token Elimination Algorithm

1. *Token Informativeness:*

$$I(T_j) = \sum_{i \neq j} S_{ij} = \sum_{i \neq j} Q_i K_j$$

2. *Score Block Informativeness:*

$$I(\mathcal{B}_{mn}^S) = \max_{S_{ij} \in \mathcal{B}_{mn}^S} S_{ij} = \max_{S_{ij} \in \mathcal{B}_{mn}^S} \frac{\sum_{h,S_{ij}^h>0} S_{ij}^h}{N_{\text{head}}}$$

3. *Token Block Informativeness:*

$$I(B_n) = \sum_m I(B_{mn})$$

4. *Compare with layer-specific threshold θ_l.*
5. *Eliminate unnecessary token blocks.*

Design 2: Context-aware Pattern Prediction

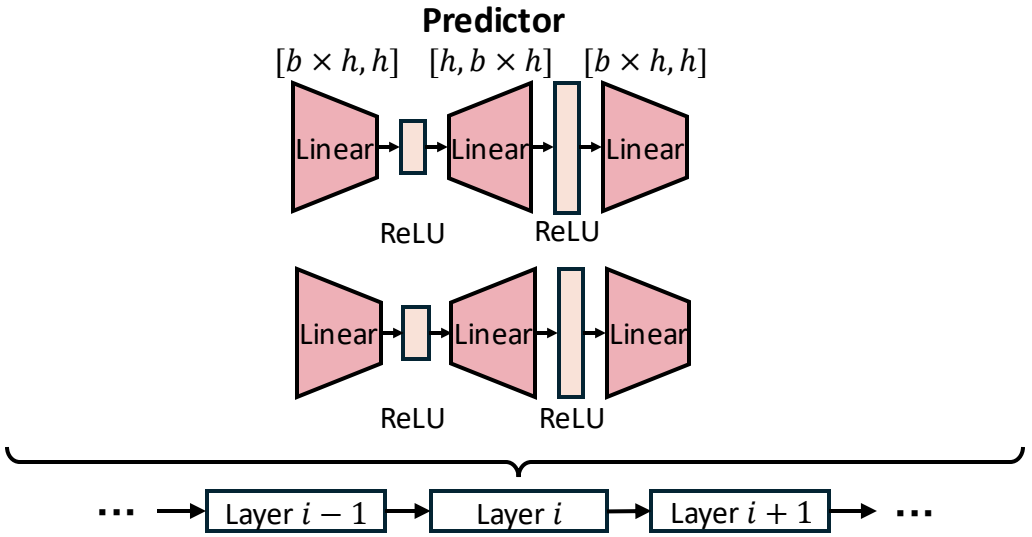
- *Deriving the exact sparsity patterns from the full attention scores is prohibitively expensive.*

Design 2: Context-aware Pattern Prediction

➢ *Deriving the exact sparsity patterns from the full attention scores is prohibitively expensive.*

- Optimization 1: Neural-Network-Based Predictor.

- JENGA deploys a pair of predictors in each layer to approximate the informativeness scores of Q and K.

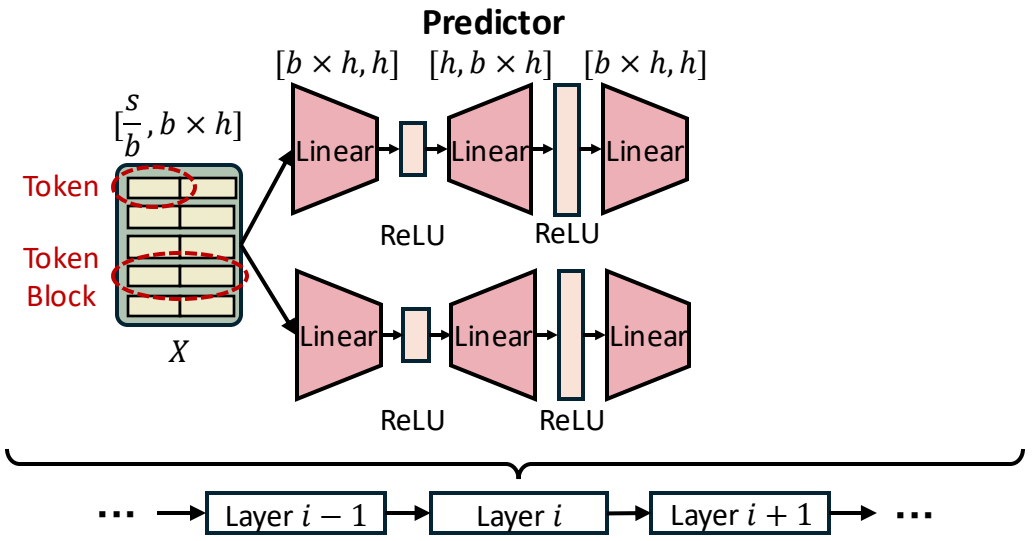


Design 2: Context-aware Pattern Prediction

➢ *Deriving the exact sparsity patterns from the full attention scores is prohibitively expensive.*

- Optimization 1: Neural-Network-Based Predictor.

- JENGA deploys a pair of predictors in each layer to approximate the informativeness scores of Q and K.

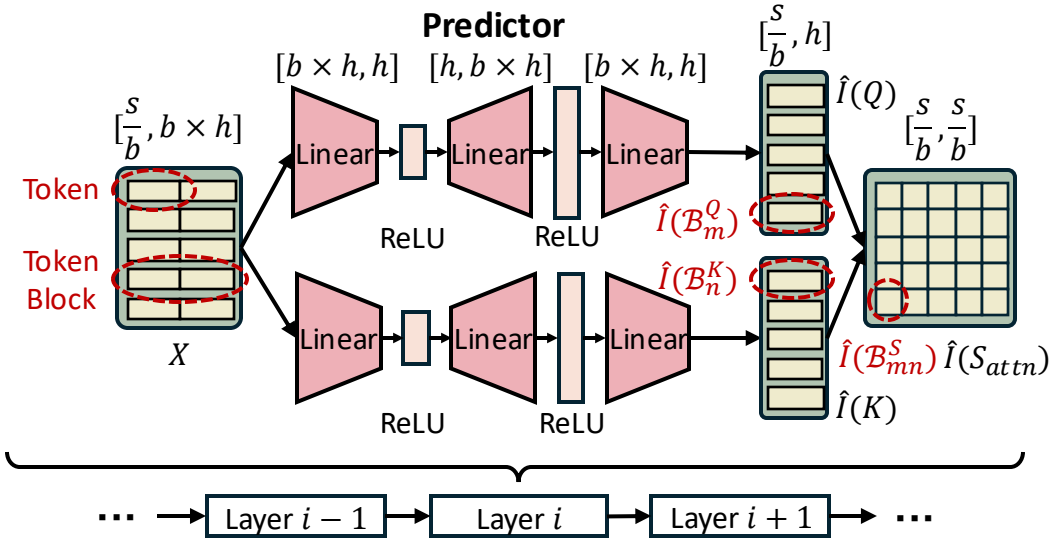


Design 2: Context-aware Pattern Prediction

➢ *Deriving the exact sparsity patterns from the full attention scores is prohibitively expensive.*

- Optimization 1: Neural-Network-Based Predictor.

- JENGA deploys a pair of predictors in each layer to approximate the informativeness scores of Q and K.



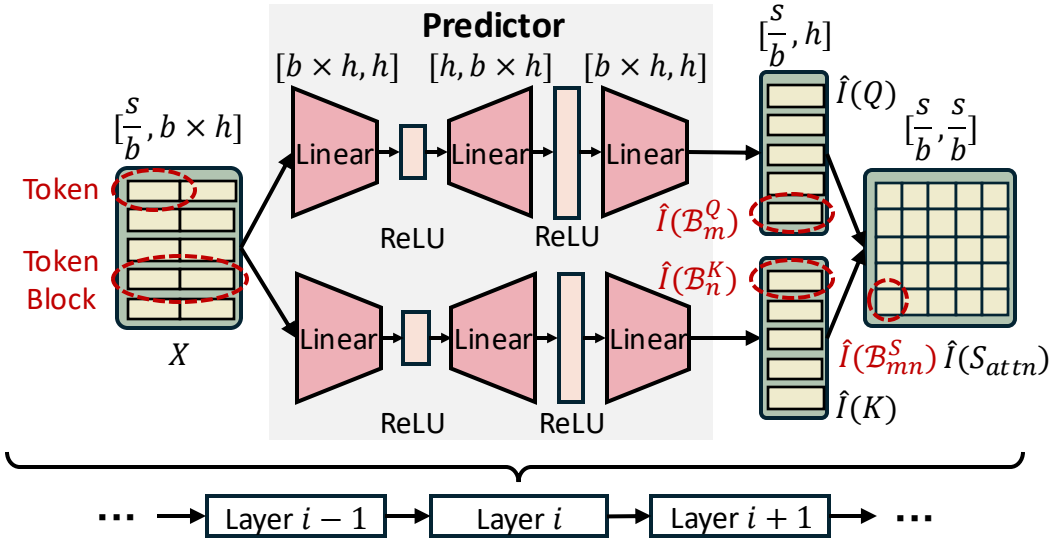
$$\hat{I}(S_{\text{attn}}) = \hat{I}(Q)\hat{I}(K)^T$$

$$\hat{I}(\mathcal{B}_{mn}^S) = \hat{I}(\mathcal{B}_m^Q)\hat{I}(\mathcal{B}_n^K)^T$$

Design 2: Context-aware Pattern Prediction

➢ *Deriving the exact sparsity patterns from the full attention scores is prohibitively expensive.*

- Optimization 1: Neural-Network-Based Predictor.
- JENGA deploys a pair of predictors in each layer to approximate the informativeness scores of Q and K.
- With a limited training dataset, these predictors can quickly converge and perform well in prediction.



$$\hat{I}(S_{\text{attn}}) = \hat{I}(Q)\hat{I}(K)^T$$

$$\hat{I}(\mathcal{B}_{mn}^S) = \hat{I}(\mathcal{B}_m^Q)\hat{I}(\mathcal{B}_n^K)^T$$

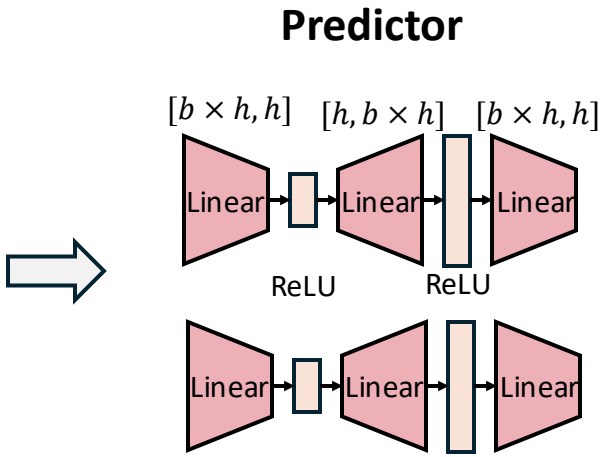
Design 2: Context-aware Pattern Prediction

➢ *The optimal sparsity patterns can only be determined at runtime, varying across different inputs.*

Dynamic nature of contextual token sparsity.

Predict the optimal sparsity patterns at runtime.

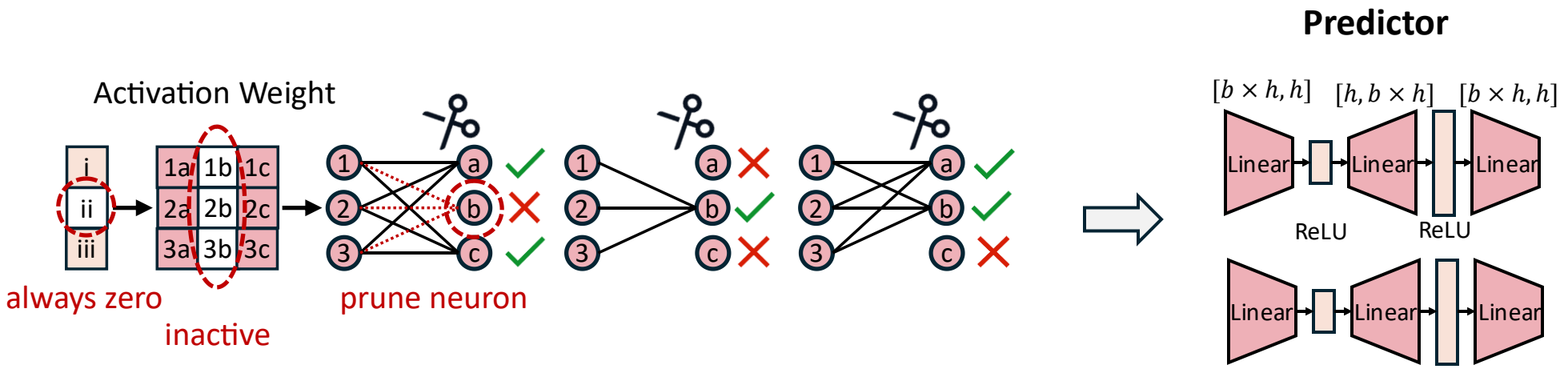
Reduce the overhead introduced by predictors.



Design 2: Context-aware Pattern Prediction

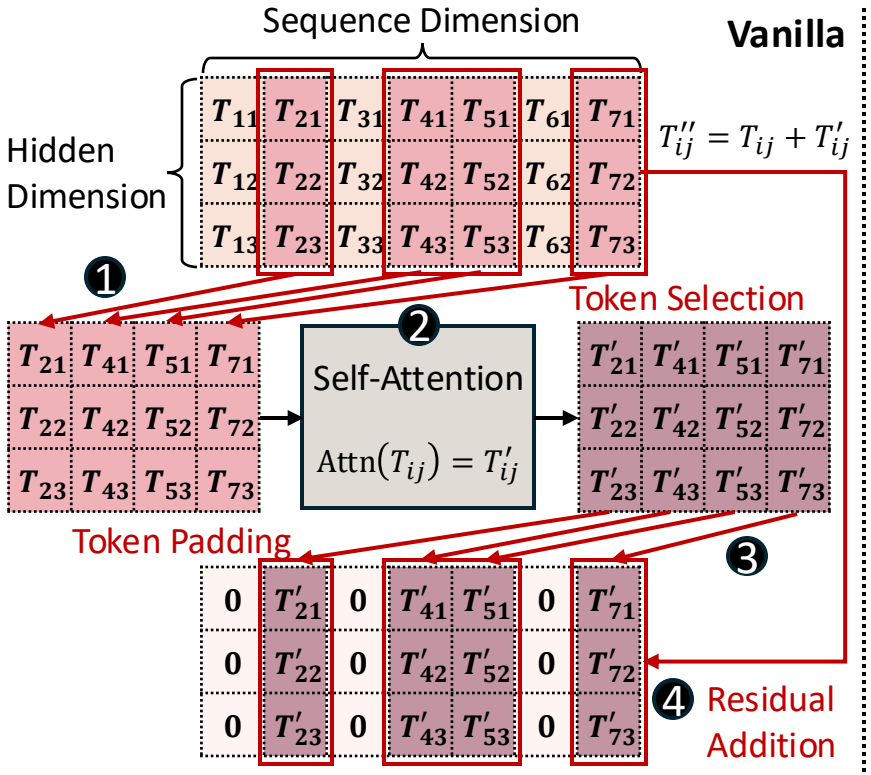
➢ *The optimal sparsity patterns can only be determined at runtime, varying across different inputs.*

- Optimization 2: Elastic Size Transformation.
- JENGA dynamically prunes neurons in the predictors based on their individual sparsity characteristics.
- Determines the optimal size for each predictor, reducing both computational and memory overhead.



Design 3: High-performance Kernel Optimization

➢ *The variability in sparsity patterns across layers necessitates iterative token selection and padding.*



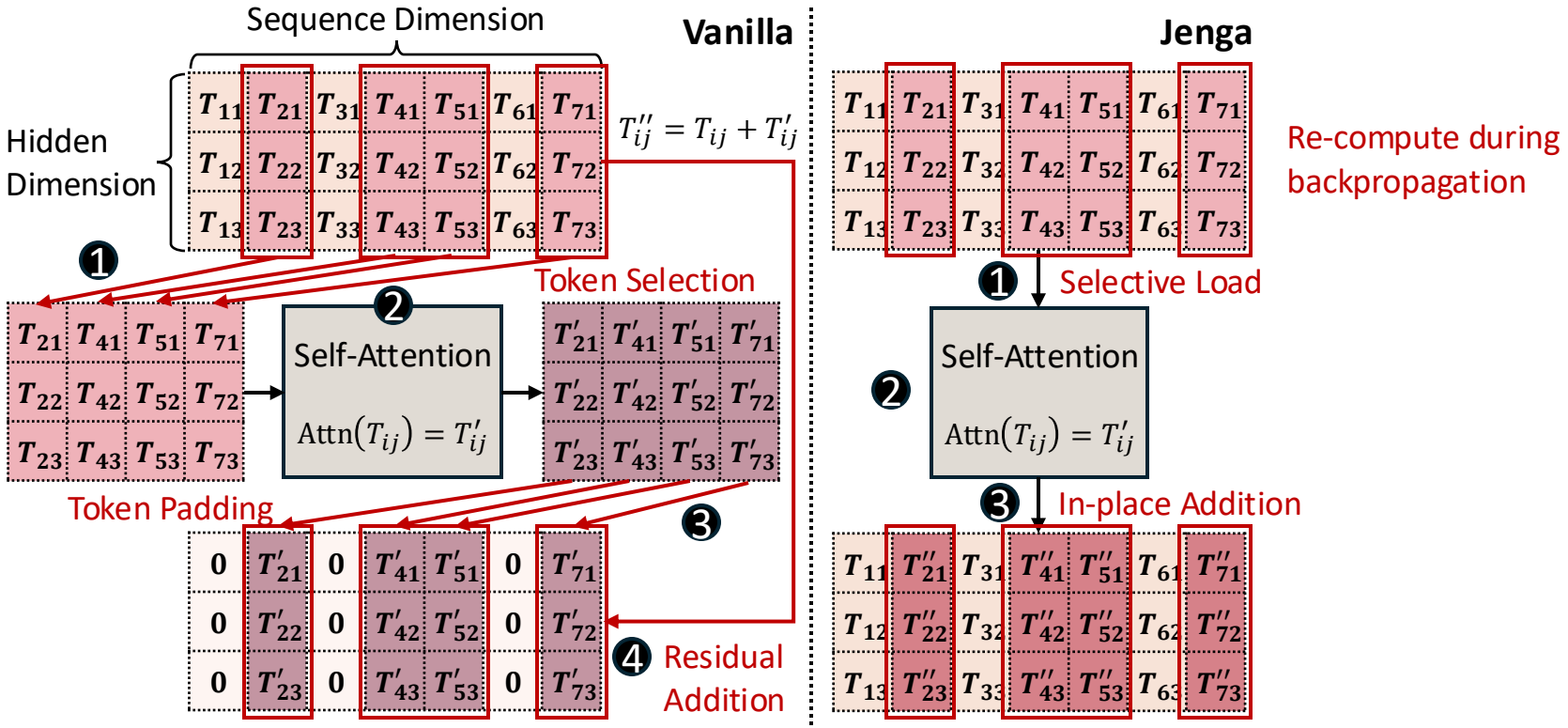
Step 1 and 3 involve extensive data movement in global memory, which incurs high memory access latency!

Design 3: High-performance Kernel Optimization

➢ *The variability in sparsity patterns across layers necessitates iterative token selection and padding.*

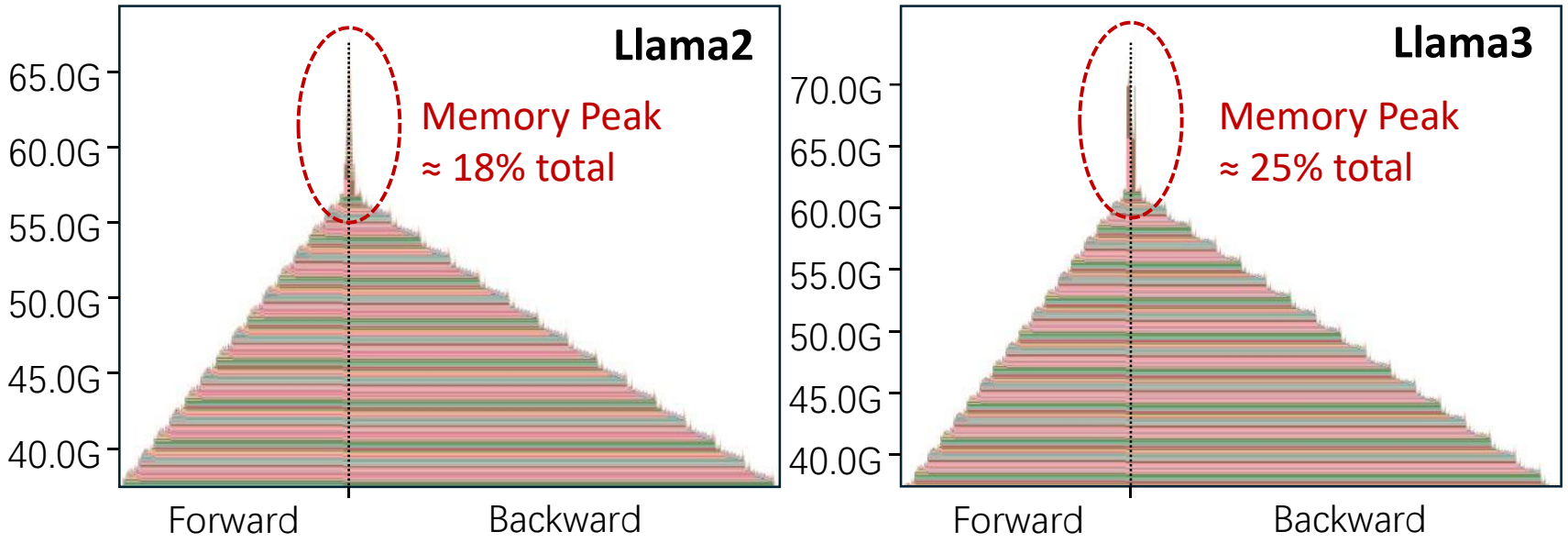
- Optimization 1: Permutation-Free Token Movement.

- JENGA fuses all unnecessary permutation operations with the attention computations.



Design 3: High-performance Kernel Optimization

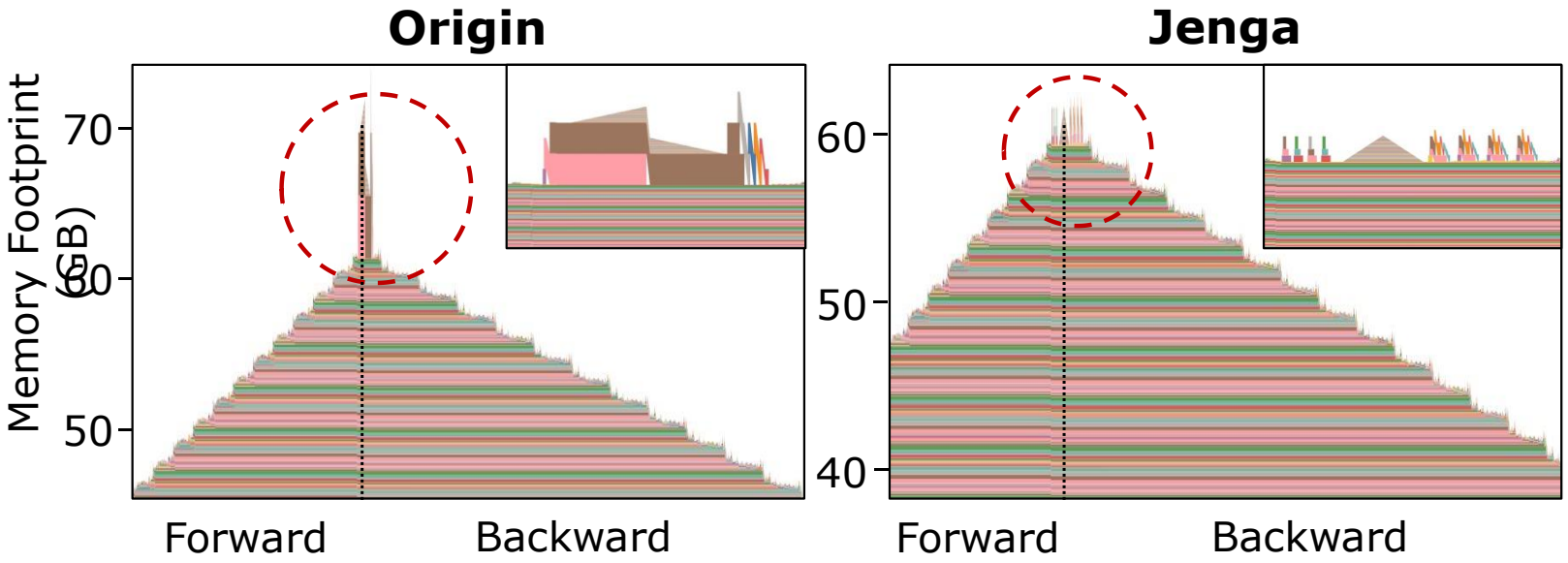
- *The extensive vocabulary size necessitates substantial activation memory to compute loss gradient.*



Design 3: High-performance Kernel Optimization

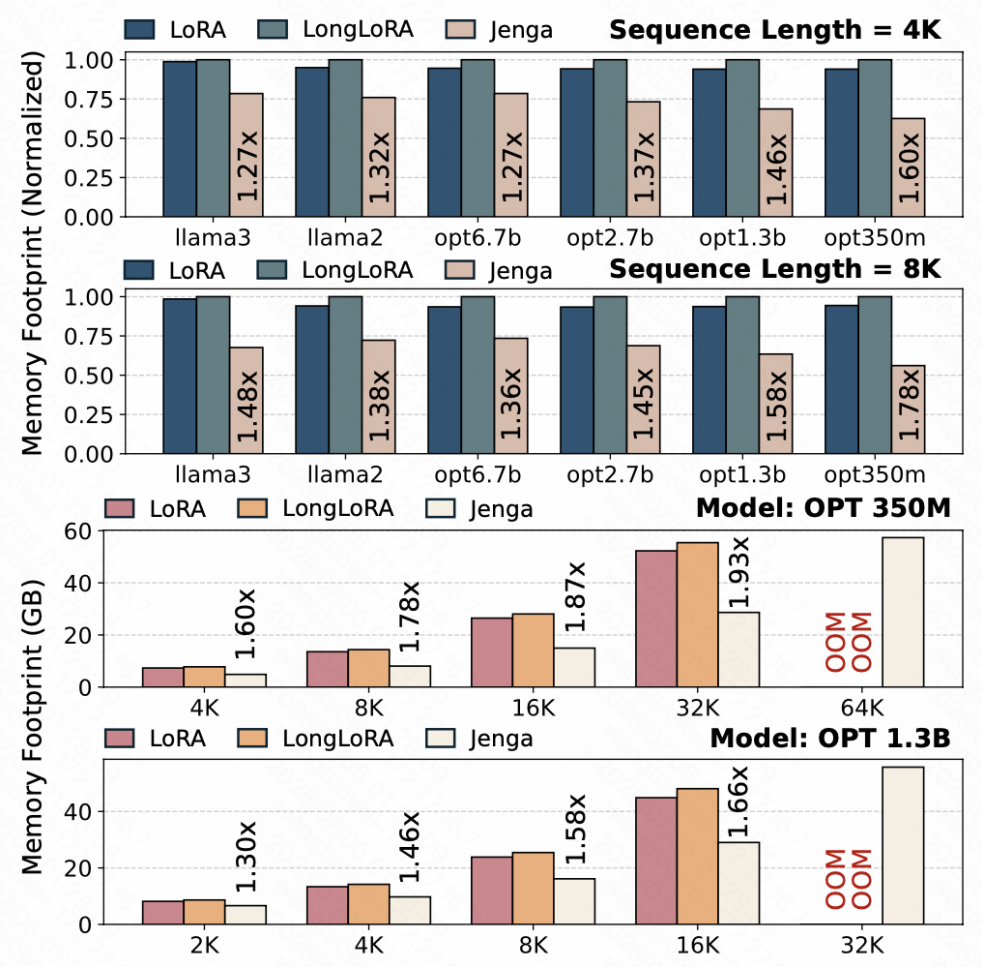
➢ *The extensive vocabulary size necessitates substantial activation memory to compute loss gradient.*

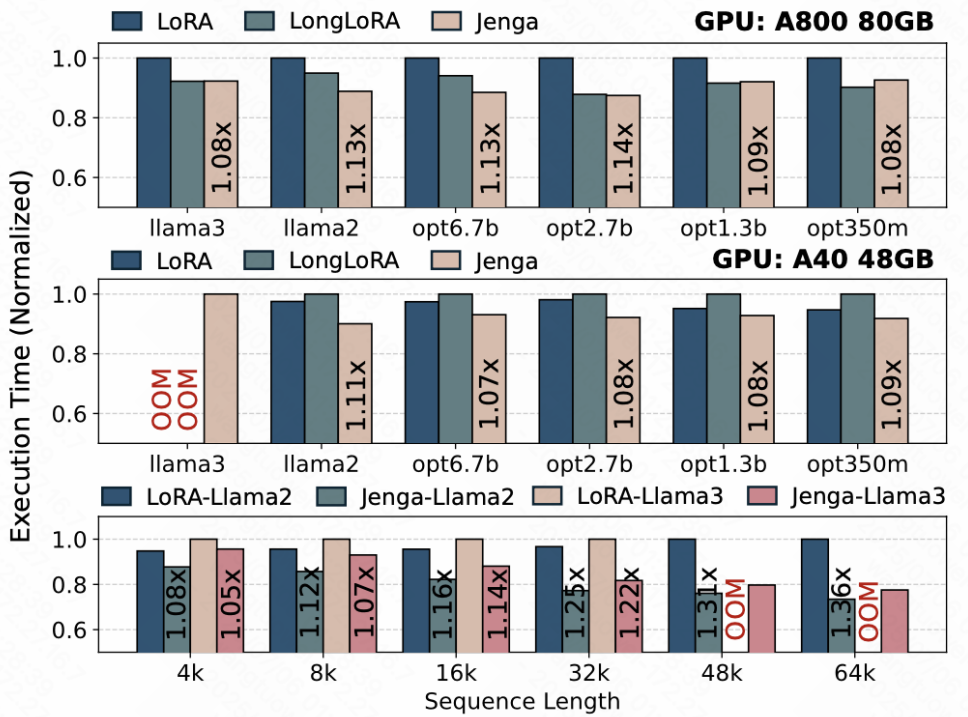
- Optimization 2: Segment-Based Peak Cutting.
- JENGA partitions the token sequence into smaller segments during final loss computation.
- The activation memory peak is reduced to $1/N$ when the sequence is divided into N segments.



EVALUATION

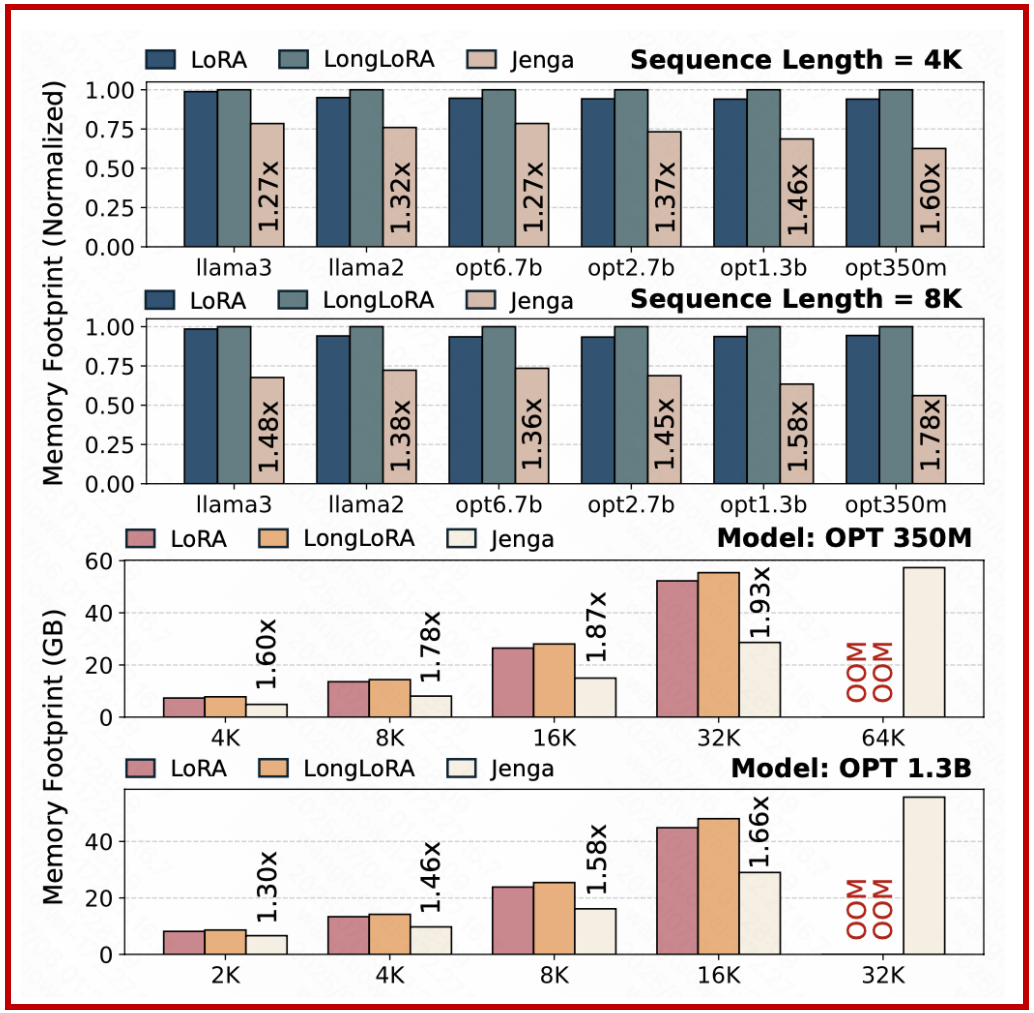
Evaluation – Memory, Speedup and Accuracy

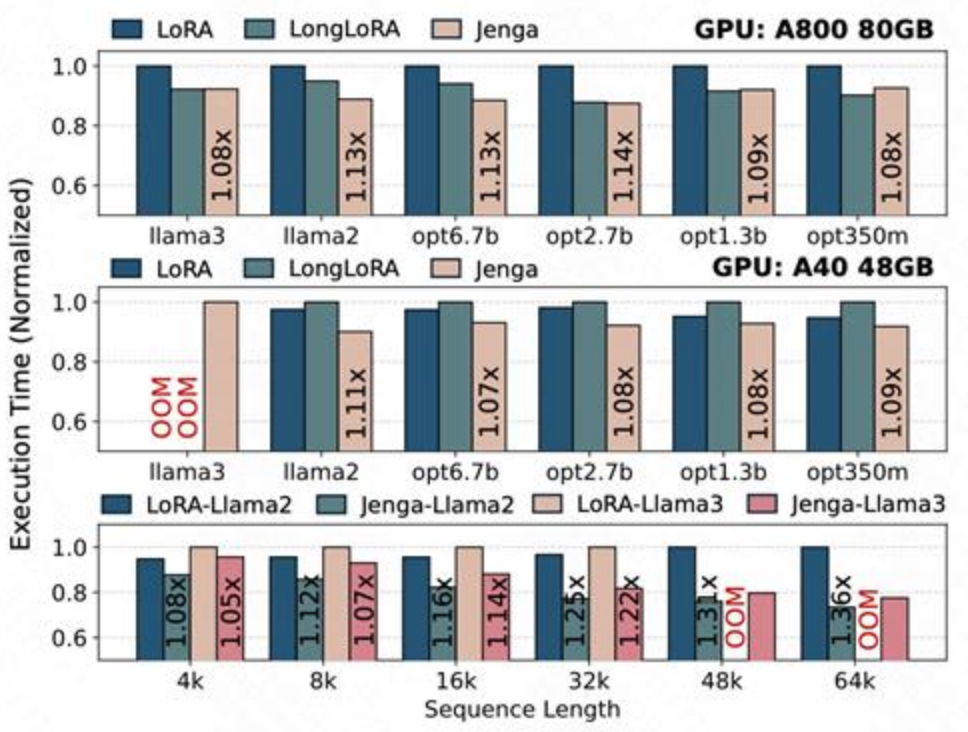




Tasks	mfqa_zh	mfqa_en	gov_report	triviaqa	vcsum	qmsum	musique	2wikimqa	repobench
Origin	23.45	23.22	**27.44**	**84.60**	**13.30**	**22.64**	4.63	9.01	**52.00**
Ours	**23.53**	**24.74**	25.92	82.59	13.02	20.33	**5.73**	**10.14**	48.32
Tasks	qasper	hotpotqa	multi_news	pr_zh	pr_en	trec	lsht	dureader	lcc
Origin	15.94	9.40	**24.43**	**10.0**	20.0	**68.0**	21.0	**23.69**	**71.28**
Ours	**17.68**	**9.55**	22.53	8.00	**22.0**	**68.0**	**25.0**	21.37	70.32

Evaluation – **Memory**, Speedup and Accuracy

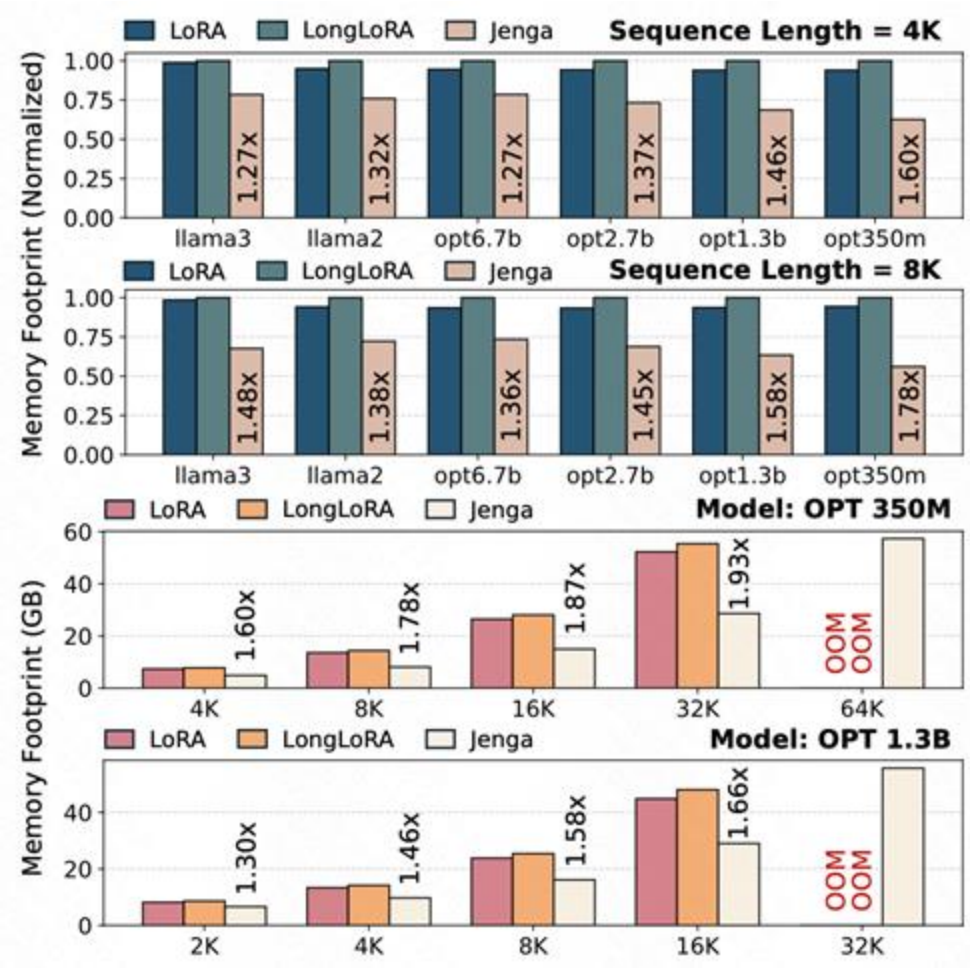


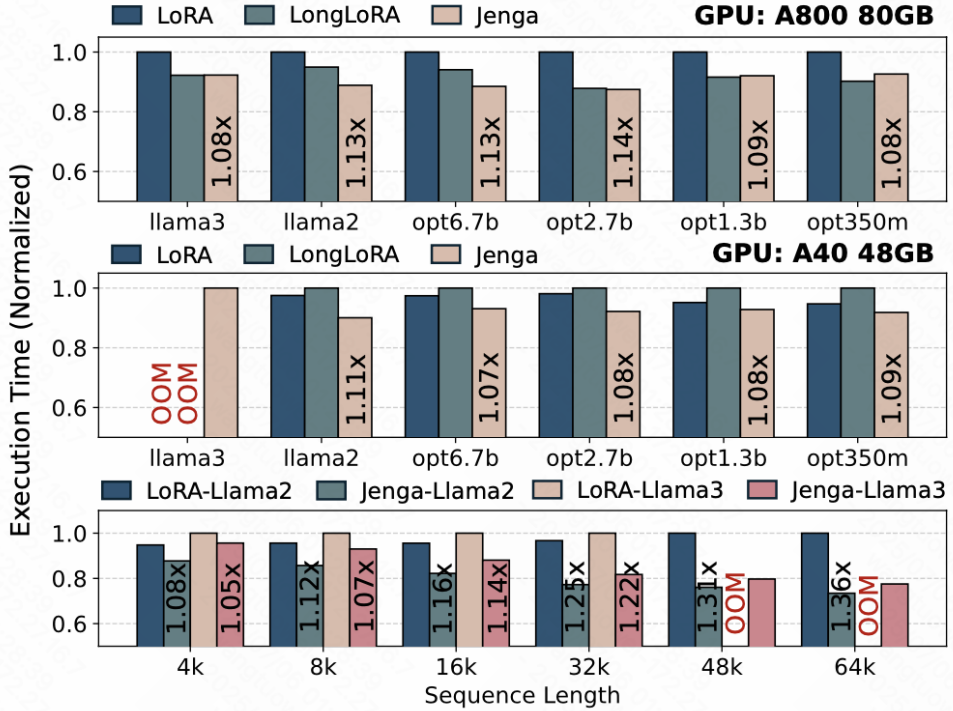


Tasks	mfqa_zh	mfqa_en	gov_report	triviaqa	vcsum	qmsum	musique	2wikimqa	repobench
Origin	23.45	23.22	**27.44**	**84.60**	**13.30**	**22.64**	4.63	9.01	**52.00**
Ours	**23.53**	**24.74**	25.92	82.59	13.02	20.33	**5.73**	**10.14**	48.32
Tasks	qasper	hotpotqa	multi_news	pr_zh	pr_en	trec	lsht	dureader	lcc
Origin	15.94	9.40	**24.43**	**10.0**	20.0	**68.0**	21.0	**23.69**	**71.28**
Ours	**17.68**	**9.55**	22.53	8.00	**22.0**	**68.0**	**25.0**	21.37	70.32

JENGA achieves average memory savings of 38.2% and 50.5% compared to LoRA across six different models, with sequence lengths of 4K and 8K, respectively.

Evaluation – Memory, **Speedup** and Accuracy

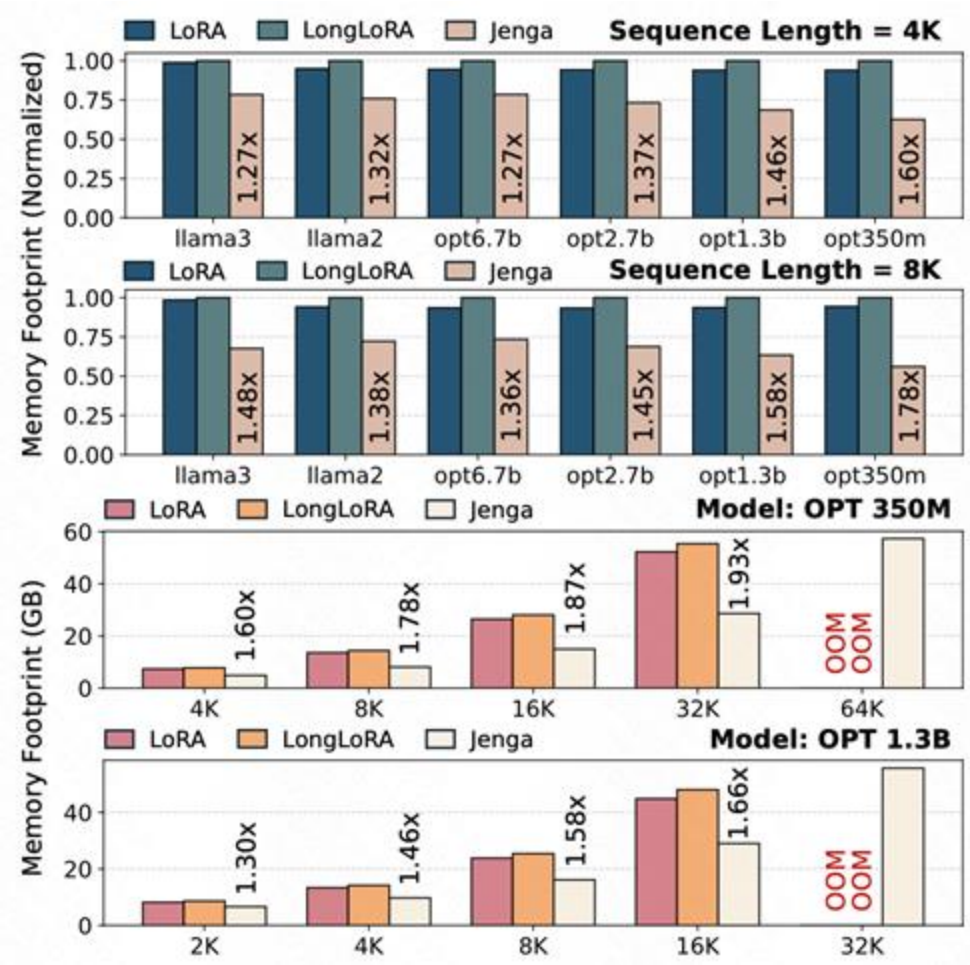


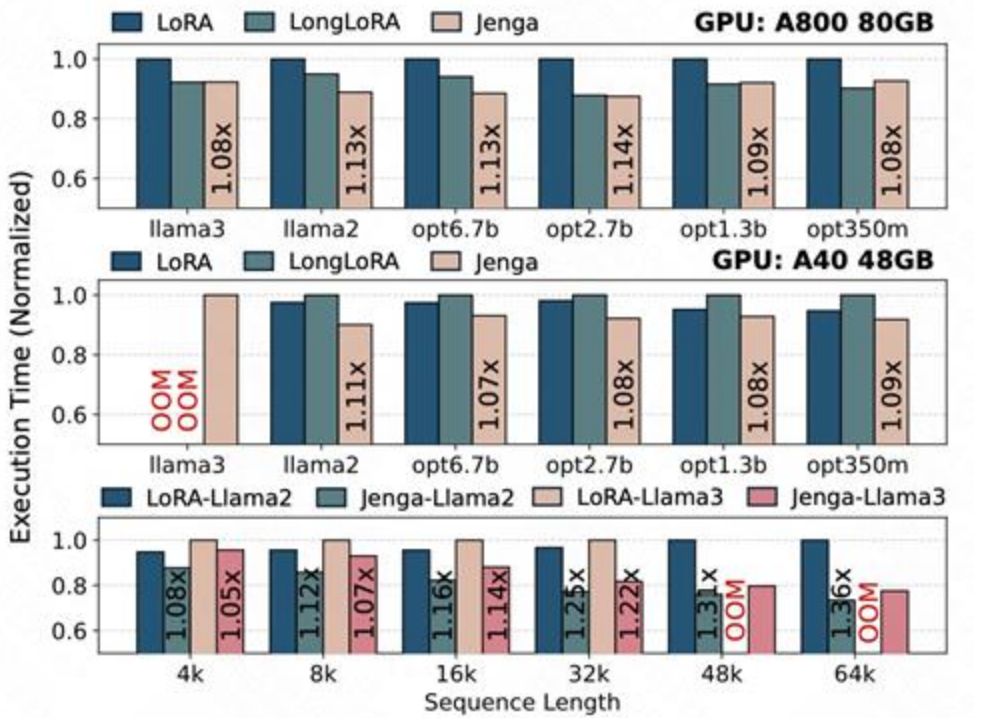


Tasks	mfqa_zh	mfqa_en	gov_report	triviaqa	vcsum	qmsum	musique	2wikimqa	repobench
Origin	23.45	23.22	**27.44**	**84.60**	**13.30**	**22.64**	4.63	9.01	**52.00**
Ours	**23.53**	**24.74**	25.92	82.59	13.02	20.33	**5.73**	**10.14**	48.32
Tasks	qasper	hotpotqa	multi_news	pr_zh	pr_en	trec	lsht	dureader	lcc
Origin	15.94	9.40	**24.43**	**10.0**	20.0	**68.0**	21.0	**23.69**	**71.28**
Ours	**17.68**	**9.55**	22.53	8.00	**22.0**	**68.0**	**25.0**	21.37	70.32

JENGA achieves average an average speedup of 10.8% and 8.6% compared to baselines on A800 and A40, respectively.

Evaluation – Memory, Speedup and **Accuracy**

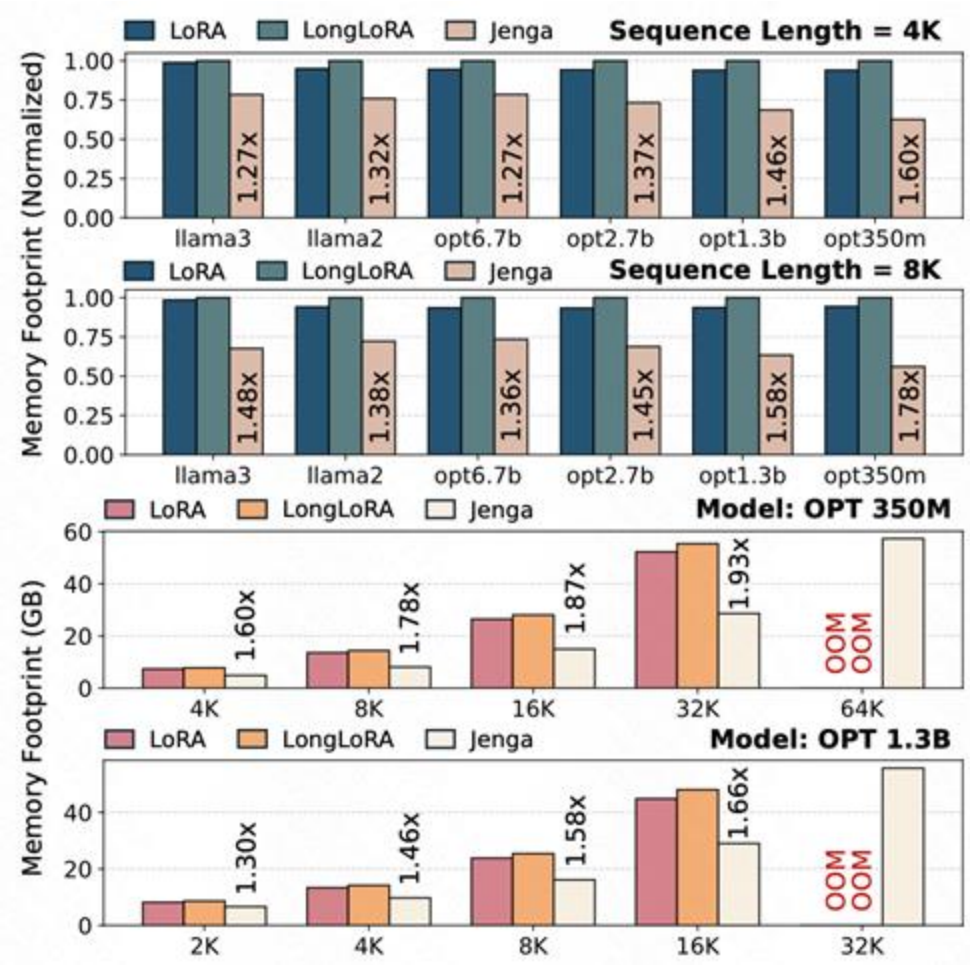


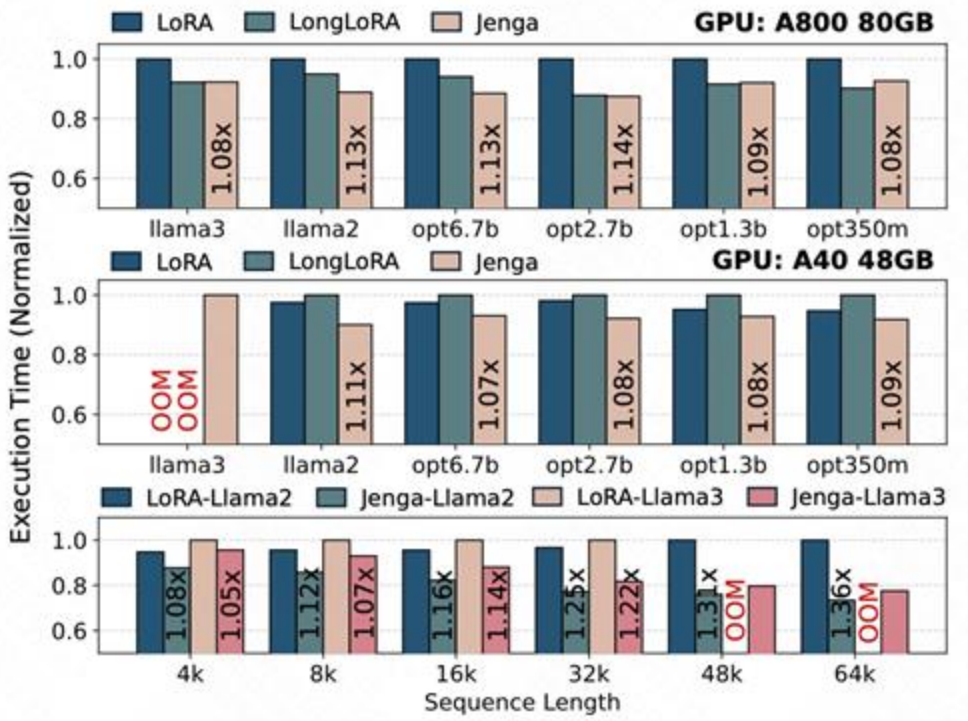


Tasks	mfqa_zh	mfqa_en	gov_report	triviaqa	vcsum	qmsum	musique	2wikimqa	repobench
Origin	23.45	23.22	**27.44**	**84.60**	**13.30**	**22.64**	4.63	9.01	**52.00**
Ours	**23.53**	**24.74**	25.92	82.59	13.02	20.33	**5.73**	**10.14**	48.32
Tasks	qasper	hotpotqa	multi_news	pr_zh	pr_en	trec	lsht	dureader	lcc
Origin	15.94	9.40	**24.43**	**10.0**	20.0	**68.0**	21.0	**23.69**	**71.28**
Ours	**17.68**	**9.55**	22.53	8.00	**22.0**	**68.0**	**25.0**	21.37	70.32

JENGA achieves accuracy comparable to the original LoRA across 18 tasks from LongBench benchmark.

Evaluation – Memory, Speedup and Accuracy





Tasks	mfqa_zh	mfqa_en	gov_report	triviaqa	vcsum	qmsum	musique	2wikimqa	repobench
Origin	23.45	23.22	**27.44**	**84.60**	**13.30**	**22.64**	4.63	9.01	**52.00**
Ours	**23.53**	**24.74**	25.92	82.59	13.02	20.33	**5.73**	**10.14**	48.32
Tasks	qasper	hotpotqa	multi_news	pr_zh	pr_en	trec	lsht	dureader	lcc
Origin	15.94	9.40	**24.43**	**10.0**	20.0	**68.0**	21.0	**23.69**	**71.28**
Ours	**17.68**	**9.55**	22.53	8.00	**22.0**	**68.0**	**25.0**	21.37	70.32

More Details in the Paper

CONCLUSION

Conclusion

- *JENGA: Enhancing LLM Long-Context Fine-Tuning with Contextual Token Sparsity*
- We identify a new sparsity mechanism, contextual token sparsity, inherent in long-context fine-tuning.
- We develop three key techniques that identify, detect, and perform contextual token sparsity.
- We envision JENGA inspiring broader exploration of sparsity for advancing LLMs.
- Repo: https://github.com/Pairshoe/Jenga-AE.

Conclusion

- *JENGA: Enhancing LLM Long-Context Fine-Tuning with Contextual Token Sparsity*
- We identify a new sparsity mechanism, contextual token sparsity, inherent in long-context fine-tuning.
- We develop three key techniques that identify, detect, and perform contextual token sparsity.
- We envision JENGA inspiring broader exploration of sparsity for advancing LLMs.
- Repo: https://github.com/Pairshoe/Jenga-AE.

Thank you !







CONTACT ME
- Tuowei Wang, CST, Tsinghua University
- Email: wtw23@mails.tsinghua.edu.cn
- Website: https://pairshoe.github.io